



TESIS - TE142599

PERANCANGAN AUTONOMOUS VTOL PADA QUADCOPTER DENGAN MENGGUNAKAN FEEDBACK LINEARIZATION DAN FUZZY TAKAGI- SUGENO

CHALIDIA NURIN HAMDANI
2213202013

DOSEN PEMBIMBING
Prof. Dr. Ir. Mohammad Nuh, DEA
Ir. Rusdhianto Effendie A.K., M.T.

PROGRAM MAGISTER
BIDANG KEAHLIAN TEKNIK SISTEM PENGATURAN
DEPARTEMEN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI ELEKTRO
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2017



TESIS - TE142599

PERANCANGAN AUTONOMOUS VTOL PADA QUADCOPTER DENGAN MENGGUNAKAN FEEDBACK LINEARIZATION DAN FUZZY TAKAGI- SUGENO

CHALIDIA NURIN HAMDANI
2213202013

DOSEN PEMBIMBING
Prof. Dr. Ir. Mohammad Nuh, DEA
Ir. Rusdhianto Effendie A.K., M.T.

PROGRAM MAGISTER
BIDANG KEAHLIAN TEKNIK SISTEM PENGATURAN
DEPARTEMEN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI ELEKTRO
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2017

LEMBAR PENGESAHAN

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar
Magister Teknik (M.T.)
di
Institut Teknologi Sepuluh Nopember

oleh:

Chalidia Nurin Hamdani
NRP. 2213202013

Tanggal Ujian : 9 Juni 2017
Periode Wisuda : September 2017

Disetujui oleh:

1. Prof. Dr. Ir. Mohammad Nuh, DEA (Pembimbing I)
NIP: 195906171984031002

2. Ir. Rusdhianto Effendie A.K., M.T. (Pembimbing II)
NIP: 195704241985021001

3. Prof. Dr. Ir. Achmad Jazidie, M.Eng (Penguji)
NIP: 195902191986101001

4. Dr. Tri Hastuti Agustinah, ST. MT. (Penguji)
NIP: 196808121994032001

Dekan Fakultas Teknologi Elektro



Dr. Tri Arief Sardjono, S.T., M.T.
NIP. 197002121995121001

Halaman ini sengaja dikosongkan

PERNYATAAN KEASLIAN TESIS

Dengan ini saya menyatakan bahwa isi keseluruhan Tesis saya dengan judul **“PERANCANGAN AUTONOMOUS VTOL PADA QUADCOPTER DENGAN MENGGUNAKAN FEEDBACK LINEARIZATION DAN FUZZY TAKAGI-SUGENO”** adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Mei 2017



Chalidia Nurin Hamdani

NRP. 2213202013

Halaman ini sengaja dikosongkan

PERANCANGAN AUTONOMOUS LANDING PADA QUADCOPTER DENGAN MENGGUNAKAN FEEDBACK LINEARIZATION DAN FUZZY TAKAGI-SUGENO

Nama mahasiswa : Chalidia Nurin Hamdani
NRP : 2213202013
Pembimbing : 1. Prof. Ir. Moh. Nuh, DEA
2. Ir. Rusdhianto Effendie A.K., M.T.

ABSTRAK

Quadcopter merupakan salah satu jenis *Unmanned Aerial Vehicle* (UAV) yang banyak digunakan sebagai objek penelitian saat ini. Dengan empat rotor sebagai penggerak utama, quadcopter mampu bergerak secara rotasional dan translasional. Kedua sistem gerak ini secara matematis tergolong nonlinier dan tidak stabil. Dalam penelitian ini, dirancang autonomous *Vertical Take-Off and Landing* (VTOL) dengan membagi sistem quadcopter berdasarkan sistem geraknya, yaitu sistem gerak rotasi dan sistem gerak translasi. Masing-masing sistem gerak dikendalikan dengan kontroler yang berbeda.

Sebagai *inner-loop* dalam sistem quadcopter, sistem gerak rotasi dituntut memiliki *settling time* lebih cepat dari sistem gerak translasi yang merupakan *outer loop*. *Input-state feedback linearization* digunakan untuk mengatasi permasalahan tersebut. Kontroler dirancang melalui pendekatan model referensi yang diperoleh dengan mengatur letak *pole* dominan yang diinginkan. Dengan persamaan *Input-state feedback linearization* yang diperoleh, sifat nonlinieritas dan interaksi dalam sistem dapat dieliminasi.

Permasalahan *tracking* dalam sistem gerak translasi diatasi dengan menggunakan fuzzy Takagi-Sugeno. Sistem direpresentasikan kembali sebagai aturan fuzzy dengan titik operasi sebagai variabel premis dan model linier sebagai konsekuensi. Dengan variabel premis yang sama, kontroler dirancang dengan skema *Parallel Distributed Compensation* (PDC). Nilai penguat kontroler yang mampu menjamin kestabilan sistem dan memenuhi performa H^∞ diperoleh dengan bantuan *Linear Matrix Inequality* (LMI).

Hasil simulasi dengan MATLAB menunjukkan bahwa sistem kontrol gerak rotasi yang dirancang mampu menstabilkan sudut *roll*, *pitch* dan *yaw* di 0 rad. Kontrol *tracking* pada *altitude* mampu menjamin kestabilan sistem dan membuat respon *altitude* mengikuti sinyal referensi yang diberikan. Selain itu, sistem kontrol juga mampu mengatasi gangguan yang diberikan dengan nilai L^2 gain sebesar 0.000539. Artinya, L^2 gain lebih kecil dari level pelemahan yang ditentukan yaitu sebesar 0.003. Hal ini menunjukkan bahwa performa H^∞ telah terpenuhi.

Kata kunci: *input-state feedback linearization*; fuzzy T-S; quadcopter; vtol.

Halaman ini sengaja dikosongkan

AUTONOMOUS VTOL DESIGN IN QUADCOPTER USING FEEDBACK LINEARIZATION AND FUZZY TAKAGI-SUGENO

By :Chalidia Nurin Hamdani
Student Identity Number :2213202013
Supervisor(s) :1. Prof. Ir. Moh. Nuh, DEA
2. Ir. Rusdhianto Effendie A.K., M.T.

ABSTRACT

Quadcopter is an Unmanned Aerial Vehicle (UAV) that is frequently used as an object of study now. Its four rotors that is the main drive, let quadcopter move in rotation and translation. Both of this motion systems are unstable dan matematically nonlinear. In this study, we have designed an autonomous *Vertical Take-Off and Landing* (VTOL) by dividing quadcopter system based on its motion systems, i.e. rotation motion system and translation motion system. Each motion system is controlled by different controller.

As the inner-loop in quadcopter system, rotation motion system is forced to have a settling time that is faster then the translation motion system as the outer loop . Input-state feedback linearization is used tosolve this problem. Reference models are obtained by set the dominant pole. Using state feedback equation that was obtained, nonlinearity and interaction inside the system can be eliminated.

Tracking problem on translation motion system is solved by using fuzzy Takagi-Sugeno. System is represented by fuzzy rules with operating points as premise variable and linear models as consequent. By using the same premise variable, the controller is designed based on Parallel Distributed Compensation (PDC) scheme. The appropriate controller gain which can stabilize system and meet the H_∞ performance is obtained with the help of Linear Matrix Inequality (LMI).

Simulation results with MATLAB show that the control system have been designed can stabilize rotation motion system at 0 rad. Altitude tracking control can make the semi-global stabil system guaranteed and altitude response is able to track the given reference signal. In addition, the control system is able to attenuate the given bounded disturbance signal with L^2 gain is about 0.000539. Its mean L^2 gain is less than the prescibed attenuation level which is 0.003. This shows that the H_∞ performance is met.

Key words: *input-statefeedback linearization*; fuzzy T-S; quadcopter; vtol.

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji syukur kehadiran Allah SWT yang senantiasa mencurahkan kasih dan sayang-Nya sehingga penulis mampu menyelesaikan Tesis dengan judul *Perancangan Autonomous VTOL pada Quadcopter dengan Menggunakan Feedback Linearization dan Fuzzy Takagi-Sugeno*. Tesis ini disusun sebagai persyaratan guna menyelesaikan Program Magister di Bidang Studi Teknik Sistem Pengaturan, Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember.

Penulis menyampaikan terima kasih kepada ayah dan ibu atas doa dan dukungan yang tidak pernah putus, Bapak Moh. Nuh dan Bapak Rusdhianto Effendie A.K. atas bimbingan, motivasi dan dukungan yang selalu diberikan selama penulis menempuh studi, Bapak dan Ibu Dosen Pengajar di Teknik Sistem Pengaturan atas didikan dan pengajaran yang diberikan, dan teman-teman Program Magister Teknik Sistem Pengaturan atas dukungan dan bantuan selama ini. Tidak lupa penulis ucapkan terima kasih kepada Kementerian Ristek dan Dikti atas beasiswa *freshgraduate* berupa bantuan dana pendidikan Program Magister.

Akhir kata, penulis berharap Tesis ini dapat bermanfaat bagi siapapun yang membutuhkan, terutama bagi penulis sendiri. Alhamdulillahirabbil'alamiin.

Surabaya, 15 Mei 2017

Penulis

Halaman ini sengaja dikosongkan

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN	iii
PERNYATAAN KEASLIAN TESIS.....	v
ABSTRAK.....	vii
KATA PENGANTAR	xi
DAFTAR ISI.....	xiii
DAFTAR GAMBAR.....	xv
DAFTAR TABEL.....	xvii
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan	4
1.4 Batasan Masalah.....	4
1.5 Kontribusi.....	5
1.6 Metodologi Penelitian	5
BAB 2 KAJIAN PUSTAKA	7
2.1 Kajian Penelitian Terkait.....	7
2.1.1 <i>Intelligent Fuzzy Controller of A Quadrotor</i>	7
2.1.2 <i>Robust Fuzzy Control for Stabilization of Quadrotor</i>	9
2.1.3 <i>Robust Control of a Quadrotor using Takagi-Sugeno</i> <i>Fuzzy Model and an LMI Approach</i>	11
2.2 Teori Dasar.....	13
2.2.1 Quanser Qball	14
2.2.2 Gerakan Dasar Quadcopter	15
2.2.3 Gaya dan Momen pada Quadcopter.....	16
2.2.4 Dinamika Quadcopter	20
2.2.5 Metode <i>FeedbackLinearization</i>	23
2.2.6 Fuzzy Takagi Sugeno (T-S)	24
2.2.7 Analisis Kestabilan Lyapunov	31

2.2.8 Performa H_{∞}	33
2.2.9 <i>LinearMatrixInequality</i> (LMI)	34
BAB 3 METODOLOGI PENELITIAN	38
3.1 Sistem Gerak Rotasi	40
3.1.1 Model Matematika Sistem Gerak Rotasi	40
3.1.2 Kontroler <i>Linear State Feedback</i>	41
3.1.3 Persamaan <i>Input-StateFeedback</i>	41
3.2 Sistem Gerak Translasi	42
3.2.1 Perancangan Kontroler untuk <i>Altitude</i>	42
3.2.2 Perancangan Kontroler untuk Posisi	51
BAB 4 HASIL DAN PEMBAHASAN	59
4.1 Simulasi Sistem Kontrol dalam Kondisi Tanpa Gangguan	59
4.2 Simulasi Sistem Kontrol dengan Tambahan Gangguan Terbatas	63
BAB 5 PENUTUP	71
5.1 Kesimpulan	71
5.2 Saran	71
DAFTAR PUSTAKA	73
LAMPIRAN.....	77

DAFTAR GAMBAR

Gambar 2.1 Diagram blok sistem kontrol quadcopter secara keseluruhan.....	7
Gambar 2.2 Respon height terhadap sinyal step dari 6 ke 8 m.....	8
Gambar 2.3 Respon <i>roll</i> terhadap sinyal step dari 0 ke -0.5 rad.....	8
Gambar 2.4 Respon yaw terhadap sinyal step dari 0 ke -0.5 rad.....	9
Gambar 2.5 Respon sudut roll, pitch dan yaw	10
Gambar 2.6 Respon kecepatan sudut roll, pitch dan yaw	10
Gambar 2.7 Diagram blok sistem kontrol quadcopter	11
Gambar 2.8 Simulasi tracking posisi x, y dan z.....	13
Gambar 2.9 Simulasi tracking attitude.....	13
Gambar 2.10 Quanser Qball X4.....	14
Gambar 2.11 Gerakan <i>Throttle</i>	15
Gambar 2.12 Gerakan <i>Roll</i>	15
Gambar 2.13 Gerakan <i>Pitch</i>	15
Gambar 2.14 Gerakan <i>yaw</i>	15
Gambar 2.15 E-frame dan B-frame	17
Gambar 2.16 Rotasi B-frame terhadap sumbu <i>X</i> pada E-frame.....	18
Gambar 2.17 Rotasi B-frame terhadap sumbu <i>Y</i> pada E-frame.....	19
Gambar 2.18 Rotasi B-frame terhadap sumbu <i>Z</i> pada E-frame.....	20
Gambar 2.19 (a) himpunan klasik, (b) himpunan fuzzy	25
Gambar 2.20 Fungsi segitiga	26
Gambar 2.21 Fungsi trapesium.....	26
Gambar 2.22 Fungsi Gaussian	27
Gambar 2.25 Fungsi <i>GeneralizedBell</i>	27
Gambar 2.24 Struktur umum sistem inferensi fuzzy	29
Gambar 2.25 Kontrol H_{∞}	33
Gambar 3.1 Diagram blok sistem kontrol pada quadcopter	38
Gambar 3.2 Diagram blok sistem gerak rotasi.....	39
Gambar 3.3 Diagram blok sistem kontrol dengan fuzzy T-S.	28
Gambar 3.4 Fungsi keanggotaan.....	46

Gambar 3.5 Diagram blok sistem kontrol gerak translasi di sumbu x	54
Gambar 3.6 Diagram blok sistem kontrol gerak translasi di sumbu y	56
Gambar 1.1 Respon sistem gerak rotasi terhadap sinyal <i>unitstep</i> : (a) gerak <i>roll</i> , (b) gerak <i>pitch</i> dan (c) gerak <i>yaw</i>	60
Gambar 4.2 Respon <i>tracking</i> pada <i>altitude</i>	61
Gambar 4.3 Respon kecepatan vertikal	61
Gambar 4.4 Respon posisi (a) posisi x dan (b) posisi y	62
Gambar 4.5 Respon <i>tracking</i> pada <i>altitude</i> dengan perubahan posisi x dan y	62
Gambar 4.5 Noise berupa <i>randomnumber</i> dengan <i>mean</i> 0 dan <i>variance</i> 0.001	63
Gambar 4.5 Respon sudut <i>roll</i> dan posisi y ketika diberikan <i>noise</i> pada sudut <i>roll</i> dan <i>pitch</i>	64
Gambar 4.6 Respon sudut <i>pitch</i> dan posisi x ketika diberikan <i>noise</i> pada sudut <i>roll</i> dan <i>pitch</i>	64
Gambar 4.7 Respon <i>tracking</i> pada <i>altitude</i> ketika diberikan <i>noise</i> pada sudut <i>roll</i> dan <i>pitch</i>	64
Gambar 4.8 Noise berupa <i>randomnumber</i> dengan <i>mean</i> 0 dan <i>variance</i> 0.01	65
Gambar 4.9 Respon <i>tracking</i> pada <i>altitude</i> dengan pemberian <i>noise</i> pada keluaran <i>altitude</i>	65
Gambar 4.10. Respon kecepatan vertikal saat diberikan <i>noise</i> pada keluaran <i>altitude</i>	66
Gambar 4.11 Respon <i>tracking</i> pada <i>altitude</i> dengan <i>noise</i> dan $\rho = 0.05$	68
Gambar 4.12. Sinyal gangguan.....	68
Gambar 4.12 Respon <i>tracking</i> pada <i>altitude</i> dengan penambahan gangguan.....	69
Gambar 4.4 Respon sistem kontrol gerak translasi tanpa gangguan.	39

DAFTAR TABEL

Tabel 2.1 Parameter Quadcopter.....	23
Tabel 4.1 Nilai ISE <i>altitude</i> pada simulasi dengan berbagai nilai <i>noise</i>	66
Tabel 4.2 Daftar nilai K_i untuk berbagai nilai ρ	67
Tabel 4.3 Hasil simulasi saat kondisi ada gangguan dengan berbagai nilai ρ	69

Halaman ini sengaja dikosongkan

BAB 1

PENDAHULUAN

1.1 ++Latar Belakang

Pada dekade terakhir, dunia robotika mengalami perkembangan pesat dengan hadirnya kendaraan udara tak berawak atau yang biasa disebut *Unmanned Aerial Vehicle* (UAV). Penggunaan UAV ini dapat dikategorikan cukup luas seperti untuk keperluan militer, keamanan, dan yang lainnya.[1]

Salah satu jenis UAV yang banyak diteliti saat ini adalah quadcopter, UAV jenis mini helicopter yang menggunakan empat motor sebagai penggerak utamanya. Sisi nonlinearitas dan *Multiple Input-Multiple Output* (MIMO) dari model quadcopter ini menjadi tantangan tersendiri untuk terus diteliti.

Dalam pengoperasian quadcopter, salah satu mekanisme dasar yang harus dilakukan adalah *Vertical Take-Off and Landing* (VTOL). Untuk dapat melakukan mekanisme ini, proses pertama yang harus diperhatikan adalah stabilisasi. Semua sudut quadcopter terhadap sumbu x , y dan z yang selanjutnya disebut sudut *roll*, *pitch* dan *yaw* harus distabilkan di titik 0 (nol) radian. Saat ini sudah cukup banyak penelitian yang membahas tentang stabilisasi sudut quadcopter. [2] telah berhasil menstabilkan sudut quadcopter dengan metode fuzzy mamdani walaupun masih terdapat *overshoot* dalam respon sistem lup tertutup yang dirancang. Dengan kontroler yang berbeda yaitu kontrol H_∞ , [3] juga mampu menstabilkan sudut quadcopter. Namun, dari hasil simulasi yang dilakukan, waktu yang dibutuhkan untuk menstabilkan sudut quadcopter masih relatif lama, yaitu 30 detik. Waktu stabilisasi sudut quadcopter penting diperhatikan karena masalah stabilisasi ini merupakan *inner loop* dalam sistem quadcopter. Dalam kasus ini, *inner loop* harus memiliki *settling time* lebih cepat dari *outer loop* sehingga *outer loop* dapat dikendalikan dengan baik,.

Pendekatan yang berbeda digunakan [4] untuk mengatasi stabilisasi quadcopter. Sistem gerak rotasi pada quadcopter direpresentasikan kembali dalam beberapa model linier yang disusun sebagai aturan fuzzy. Linierisasi dilakukan dengan mendapatkan matriks Jacobian pada titik operasi yang ditentukan.

Pendekatan linier ini memang akan memudahkan perancangan kontroler, tetapi metode dalam penelitian tersebut belum cukup jika diterapkan pada sistem gerak rotasi quadcopter yang di dalamnya terdapat interaksi antar subsistem. Metode *dynamical decoupling* digunakan pada [5] untuk menghilangkan interaksi antar subsistem dalam sistem *airship* sehingga subsistem dapat dikendalikan secara independen.

Metode yang lebih sederhana digunakan oleh [6]. *Input-output linearization* digunakan untuk melinierisasi dan menghilangkan interaksi dalam sistem quadcopter. Penerapan metode ini membutuhkan pengetahuan tentang hubungan antara output dan input sistem. Untuk mereduksi hal tersebut, metode lain yang dapat digunakan adalah *input-state linearization*. Beberapa hasil penelitian lain tentang stabilisasi quadcopter dengan metode yang berbeda dibahas dalam [7] – [12].

Penelitian yang membahas secara khusus tentang mekanisme VTOL tidak sebanyak penelitian tentang stabilisasi. Namun, pada dasarnya mekanisme VTOL dapat dirancang dengan membuat sistem kontrol *tracking* pada gerak translasi sumbu z (*altitude*) sehingga quadcopter dapat mengikuti sinyal referensi yang diberikan. Cukup banyak penelitian yang membahas permasalahan *tracking* pada quadcopter. Seperti dalam [13] dan [14] yang merancang kontrol *tracking* dengan menggunakan metode kontrol nonlinier yaitu *backstepping* dan *nonlinear output feedback*. Metode kontrol nonlinier memang lebih bisa mendekati karakteristik sistem nonlinier. Namun, proses perhitungannya lebih rumit karena dibutuhkan pengetahuan yang utuh mengenai kompleksitas sistem itu sendiri [28].

Strategi lain yang dapat digunakan adalah dengan pendekatan linier lokal seperti yang digunakan pada [15]. Sistem kontrol *tracking* dirancang dengan merepresentasikan kembali sistem quadcopter dalam model fuzzy T-S yang pertama kali didesain oleh [16]. Metode *pole placement* yang diformulasikan dalam LMI digunakan untuk menjamin kestabilan sistem secara global. Penelitian ini belum lengkap karena belum menyertakan gangguan dalam sistem yang dirancang. Padahal dalam pengoperasian quadcopter, sering ditemui adanya gangguan seperti angin dan pengaruh cuaca lainnya.

Dalam sistem kontrol *tracking*, objektif yang ingin dicapai adalah *tracking error* yang sekecil mungkin baik dengan maupun tanpa adanya gangguan yang dapat mempengaruhi sistem. Untuk itu, diperlukan sistem kontrol yang diformulasikan dengan teori kekokohan sistem. Pada [18] disajikan desain kontrol *tracking* yang diformulasikan dengan performa H^∞ .

Berdasarkan latar belakang tersebut, dalam penelitian ini diajukan desain sistem kontrol untuk autonomous VTOL pada quadcopter dengan membagi sistem berdasarkan permasalahan yang dihadapi. *Input-state feedback linearization* digunakan untuk mengatasi permasalahan regulator pada sistem gerak rotasi dan fuzzy Takagi-Sugeno (T-S) digunakan untuk mengatasi permasalahan *tracking* pada sistem gerak translasi sumbu z (*altitude*). Berbeda dengan [15], kestabilan sistem akan dijamin dengan memformulasikan teori kestabilan Lyapunov pada kontroler yang dirancang. Selain itu, untuk mengatasi hadirnya gangguan, digunakan performa H^∞ sehingga sehingga dijamin diperoleh *tracking error* sekecil mungkin.

1.2 Rumusan Masalah

Quadcopter memiliki 6 derajat kebebasan yaitu *height motions*, *vertical motions*, *horizontal motions*, *pitch*, *roll* dan *yaw*. 6 derajat kebebasan ini berasal dari variasi kecepatan keempat rotor yang terdapat pada *quadcopter* sehingga didapatkan model matematika *quadcopter* yang bersifat MIMO, nonlinier dan tidak stabil.

Dari hasil analisis terhadap permasalahan dasar dalam sistem kontrol quadcopter, diperoleh hipotesa bahwa sistem kontrol quadcopter dapat dibagi menjadi 2 sistem utama dengan permasalahan yang berbeda yaitu: sistem gerak rotasi dan sistem gerak translasi.

Mengacu pada mekanisme *autonomous* VTOL yang akan dirancang, dapat dilakukan klasifikasi permasalahan pada masing-masing sistem. Dalam sistem gerak rotasi, permasalahan yang dihadapi adalah permasalahan regulator di mana sudut *roll*, *pitch* dan *yaw* harus distabilkan di 0 rad. Sebagai *inner-loop* dalam sistem quadcopter, sistem gerak rotasi dituntut memiliki *settling time* yang cepat untuk mendukung sistem gerak translasi sebagai *outer-loop*. Sedangkan dalam

sistem gerak translasi, khususnya sistem gerak translasi di sumbu z , permasalahan yang muncul adalah *tracking*. Sistem gerak translasi harus mampu mengikuti sinyal referensi yang diberikan dan mampu mengatasi gangguan yang diberikan.

Berdasarkan hal tersebut di atas, permasalahan yang akan diselesaikan dalam penelitian ini adalah bagaimana merancang sebuah sistem kontrol yang mampu memenuhi kriteria untuk masing-masing sistem, yaitu:

1. Sistem gerak rotasi dapat distabilkan di 0 rad dengan *settling time* yang cepat, dan stabil secara semi-global;
2. Sistem gerak translasi sumbu z (*altitude*) dapat mengikuti sinyal referensi yang diberikan, stabil secara semi-global dan mampu mengatasi gangguan terbatas yang diberikan dengan nilai kesalahan *tracking* yang sekecil mungkin.

1.3 Tujuan

Penelitian ini dilakukan untuk memperoleh desain sistem kontrol yang mampu mengakomodasi permasalahan-permasalahan yang telah disebutkan pada bagian rumusan masalah sehingga quadcopter dapat melakukan autonomous VTOL dengan aman. Tujuan khusus yang akan dicapai adalah:

1. Memperoleh sistem lup tertutup pada gerak rotasi yang stabil dan memiliki respon sesuai karakteristik yang diinginkan.
2. Memperoleh sistem lup tertutup pada gerak translasi sumbu z yang stabil, mampu mengikuti sinyal referensi yang diberikan dan memenuhi performa H_∞ .

1.4 Batasan Masalah

Beberapa batasan masalah yang ditentukan dalam penelitian ini antara lain:

1. Model quadcopter yang digunakan menerapkan konfigurasi 'plus';
2. Mekanisme *autonomous* VTOL dilakukan pada x dan y (0,0); dan
3. Gangguan yang diberikan bersifat terbatas.

1.5 Kontribusi

Kontribusi dari penelitian ini adalah konsep pemecahan model sistem quadcopter berdasarkan permasalahan dasar yang dihadapi guna memperoleh desain sistem kontrol yang sesuai kebutuhan. *Input-state feedback linearization* digunakan untuk mengontrol sistem gerak rotasi sehingga sistem stabil dan memiliki respon sesuai dengan karakteristik yang diinginkan. Fuzzy T-S digunakan untuk kontrol *tracking* pada sistem gerak translasi sumbu z sehingga sistem mampu mengikuti sinyal referensi yang diberikan dengan tetap memperhatikan kestabilan sistem secara semi-global berdasarkan analisis Lypunov. Performa H^∞ digunakan untuk mengatasi gangguan terbatas yang diberikan.

1.6 Metodologi Penelitian

Metodologi yang digunakan dalam penelitian ini sebagai berikut:

1. Studi literatur

Materi yang diperlukan meliputi konsep tentang model quadcopter, *input-state feedback linearization*, fuzzy Takagi-Sugeno, Analisis kestabilan Lyapunov, performa H^∞ dan teori pendukung lainnya.

2. Perancangan dan analisis sistem

Persamaan model sistem quadcopter dibagi menjadi dua, yaitu sistem gerak rotasi dan sistem gerak translasi. Untuk sistem gerak rotasi, dirancang sistem kontrol menggunakan metode *input-state feedback linearization*. Kontroler dirancang dengan pendekatan model referensi berdasarkan penempatan *pole* dominan sesuai karakteristik respon transien yang diinginkan. Sedangkan untuk sistem gerak translasi, dirancang sistem kontrol menggunakan metode fuzzy T-S. Sistem direpresentasikan kembali dalam bentuk aturan fuzzy. Kontroler dirancang dengan skema PDC. Analisis Lyapunov dan performa H^∞ digunakan untuk menjamin kestabilan sistem secara semi-global dan mengatasi gangguan terbatas yang diberikan.

3. Pengujian dan analisis performa sistem

Untuk mengetahui hasil desain sistem kontrol yang telah dirancang, dilakukan pengujian dengan bantuan MATLAB. Pengujian dilakukan dalam kondisi ideal tanpa gangguan dan kondisi dengan gangguan terbatas. Analisa hasil pengujian dilakukan untuk mengetahui performa sistem secara keseluruhan yang meliputi kestabilan sistem, ketercapaian karakteristik respon dan performa tracking. Kestabilan sistem dianalisis berdasarkan kriteria Lyapunov sedangkan performa tracking dianalisis berdasarkan performa H_∞ .

4. Penulisan laporan Tesis

Penulisan laporan Tesis dilakukan sebagai dokumentasi dari hasil penelitian yang dilakukan. Laporan Tesis terdiri dari pendahuluan, kajian pustaka, metodologi penelitian, hasil dan pembahasan dan kesimpulan.

BAB 2

KAJIAN PUSTAKA

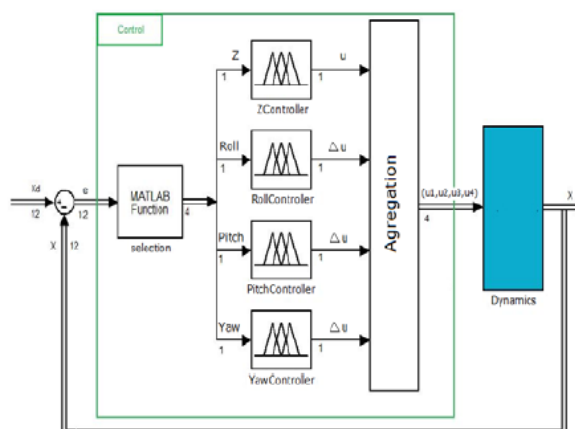
Dalam proses penyusunan penelitian ini, telah dikaji beberapa pustaka baik berupa penelitian maupun teori dasar. Pada Bab 2 ini, tidak semua pustaka akan dibahas secara mendetail, namun hanya pustaka yang menjadi dasar utama dalam penelitian ini yang akan dibahas.

2.1 Kajian Penelitian Terkait

Penelitian ini disusun berdasarkan kajian terhadap penelitian-penelitian sebelumnya. Kajian dilakukan tidak hanya pada metode kontrol saja, melainkan juga pada struktur sistem kontrol yang dirancang. Berikut adalah beberapa penelitian sebelumnya yang menjadi dasar penelitian ini.

2.1.1 *Intelligent Fuzzy Controller of A Quadrotor*

Paper ini menjelaskan tentang perancangan sistem kontrol pada quadcopter menggunakan metode fuzzy tipe mamdani. Struktur kontroler dibuat menyerupai kontroler PID di mana masukan terdiri dari *error*, *derivative error* dan *integral error*. Masing-masing masukan dikalikan dengan nilai penguat GE, GDE dan GIE. Sedangkan pada bagian keluaran dikalikan dengan penguat GU. Nilai penguat yang sesuai diperoleh dengan metode *trial and error*. Struktur ini diimplementasikan pada setiap gerak yang akan dikontrol yaitu *height*, *roll*, *pitch* dan *yaw* dengan diagram blok sebagaimana Gambar 2.1.

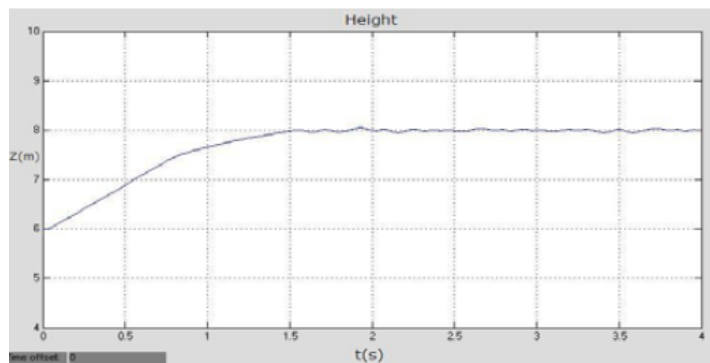


Gambar 2.1 Diagram blok sistem kontrol quadcopter secara keseluruhan

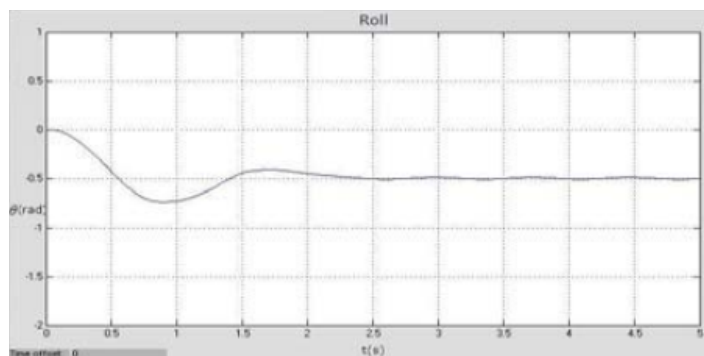
Karakteristik fuzzy controller yang dirancang adalah sebagai berikut.

1. Inferensi tipe mamdani;
2. Fungsi keanggotaan yang digunakan adalah fungsi segitiga dan trapesium;
3. Metode defuzzifikasi yang digunakan adalah centroid; dan
4. Fungsi implikasi yang digunakan adalah fungsi minimum.

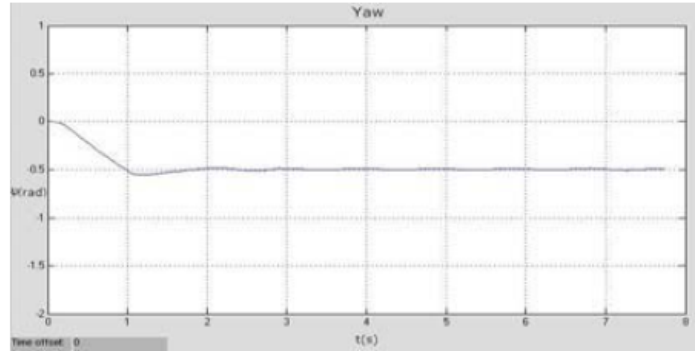
Himpunan fuzzy pada masukan menggunakan 3 fungsi keanggotaan dengan semesta pembicaraan $[-2, 2]$. Sedangkan pada bagian keluaran, himpunan fuzzy menggunakan 5 fungsi keanggotaan dengan semesta pembicaraan $[-20, 20]$. Hasil simulasi dengan MATLAB sebagaimana ditunjukkan Gambar 2.2 – 2.4.



Gambar 2.2 Respon *height* terhadap sinyal step dari 6 ke 8 m



Gambar 2.3 Respon *roll* terhadap sinyal step dari 0 ke -0.5 rad



Gambar 2.4 Respon *yaw* terhadap sinyal step dari 0 ke -0.5 rad

Respon *height* pada Gambar 2.2 menunjukkan adanya *chattering* ketika respon telah mencapai *steady state*. Pada Gambar 2.3 dan 2.4 respon *roll* dan *yaw* menunjukkan adanya *overshoot*.

Kelebihan dari kontroler ini adalah desain kontroler yang efisien dan fleksibel namun *tuning* penguat masih dilakukan secara *trial and error* sehingga relatif sulit untuk mendapatkan penguat yang sesuai, padahal penguat adalah parameter penentu dalam sistem kontrol ini. Selain itu, gangguan belum dimasukkan dalam perancangan sistem kontrolnya.

2.1.2 Robust Fuzzy Control for Stabilization of Quadrotor

Berbeda dengan paper pada 2.1.1, paper ini merancang sistem kontrol pada quadcopter dengan menggunakan metode fuzzy T-S. Sistem gerak rotasi quadcopter direpresentasikan kembali ke dalam beberapa model linier berdasarkan titik operasi yang ditentukan. Setiap model linier menggambarkan dinamika sistem nonlinier dalam titik operasi terkait.

Permasalahan kontrol yang diangkat adalah masalah stabilisasi sudut quadcopter, sehingga titik operasi ditentukan pada sudut *roll*, *pitch* dan *yaw*. Dari kombinasi 2 titik operasi yang ditentukan pada masing-masing sudut tersebut, diperoleh 8 model linier.

Model fuzzy T-S secara keseluruhan dapat dituliskan sebagaimana Persamaan (2.1)

$$\begin{aligned}\dot{x}(t) &= \sum_{i=1}^8 h_i(x) [A_i x(t) + B_i u(t)] \\ y(t) &= Cx(t)\end{aligned}\tag{2.1}$$

Kontroler dengan skema PDC dengan persamaan sinyal kontrol seperti Persamaan (2.2).

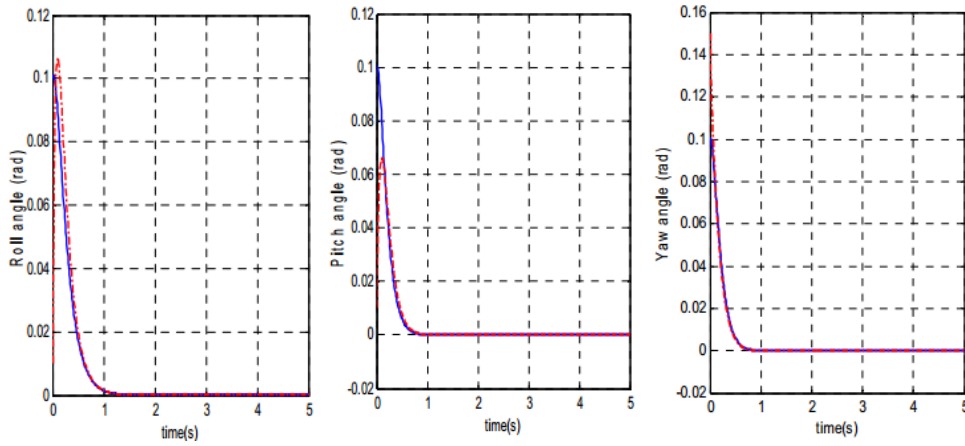
$$u(t) = \sum_{i=1}^8 h_i(z) K_i x(t)\tag{2.2}$$

Dengan mensubstitusi Persamaan (2.2) ke (2.1), maka diperoleh sistem lup tertutup sebagaimana Persamaan (2.3).

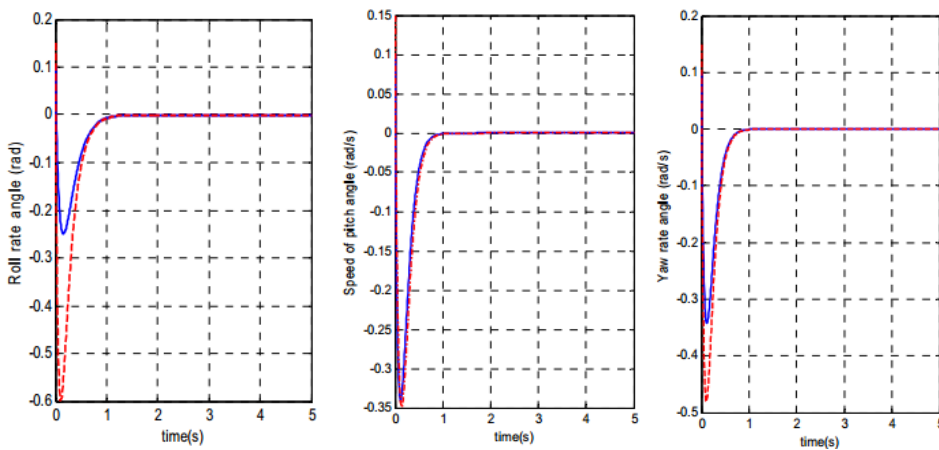
$$\begin{aligned}\dot{x}(t) &= \sum_{i=1}^8 \sum_{j=1}^8 h_i(x) h_j(x) (A_i + B_i K_j) x(t) \\ y(t) &= Cx(t)\end{aligned}\quad (2.3)$$

Kelebihan paper ini adalah adalah pendekatan linier yang digunakan. Pemodelan fuzzy T-S yang disusun berdasarkan model-model linier pada titik operasi tetap mencerminkan dinamika dari sistem quadcopter. Akan tetapi, pendekatan linier ini tidak mampu menghilangkan interaksi antar subsistem dalam sistem gerak rotasi.

Hasil simulasi dari sistem kontrol yang dirancang sebagaimana Gambar 2.5 – 2.6.



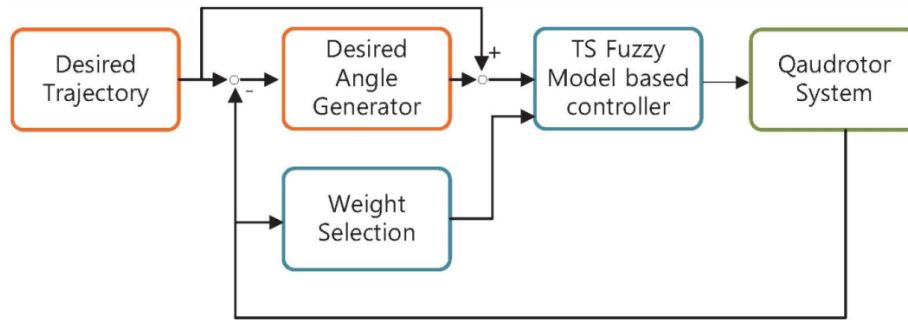
Gambar 2.5 Respon sudut *roll*, *pitch* dan *yaw*



Gambar 2.6 Respon kecepatan sudut *roll*, *pitch* dan *yaw*

2.1.3 Robust Control of a Quadrotor using Takagi-Sugeno Fuzzy Model and an LMI Approach

Jika paper sebelumnya menggunakan model fuzzy T-S untuk menyelesaikan permasalahan regulator, paper ini menggunakan pemodelan yang sama untuk menyelesaikan permasalahan *tracking*. Dalam penelitian ini, kontroler linier dirancang berdasarkan model fuzzy T-S dan pendekatan LMI. Struktur kontrol yang diajukan dalam paper ini sebagaimana ditunjukkan Gambar 2.7.



Gambar 2.7 Diagram blok sistem kontrol quadcopter.

Pada paper ini sistem quadcopter dikendalikan secara keseluruhan dan paralel. Sistem quadcopter direpresentasikan kembali dalam bentuk model fuzzy T-S dengan model secara keseluruhan dinyatakan sebagaimana Persamaan 2.4.

$$\dot{x} = \sum_{i=1}^r h_i (A_i x + B_i u) + f_g \quad (2.4)$$

dengan

$$h_i = \prod_{j=1}^p \omega_{i_j}^j$$

$$f_g = [0 \ 0 \ 0 \ 0 \ 0 \ -g \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$$

$$\sum_{i=1}^r h_i = 1$$

$$i_j \in (0,1); \ i = 1, \dots, 2^p$$

Referensi sudut *roll* dan *pitch* diperoleh dari blok *desired angle generator* dengan memanfaatkan *error tracking* gerak x dan y . Sedangkan referensi sudut *yaw* adalah 0. Blok *weight selection* digunakan untuk mendapatkan nilai

derajat keanggotaan h_i dengan menggunakan empat variabel premis yaitu *roll*, kecepatan *roll*, *pitch* dan kecepatan *pitch*. Kontroler dirancang menggunakan metode *pole placement* dan pendekatan LMI dengan persamaan sebagaimana Persamaan 2.5.

$$\delta u = \sum_{i=1}^r h_i K_i x \quad (2.5)$$

Metode *pole placement* digunakan sehingga sistem lup tertutup pada Persamaan 2.4 stabil di daerah D tertentu jika dan hanya jika terdapat matriks Q dan Y_i yang memenuhi kondisi yang dinyatakan dalam bentuk LMI sebagaimana Persamaan 2.6.

$$A_i Q + Q A_i^T + B_i Y_i + Y_i^T B_i^T < 0, i = 1, \dots, r$$

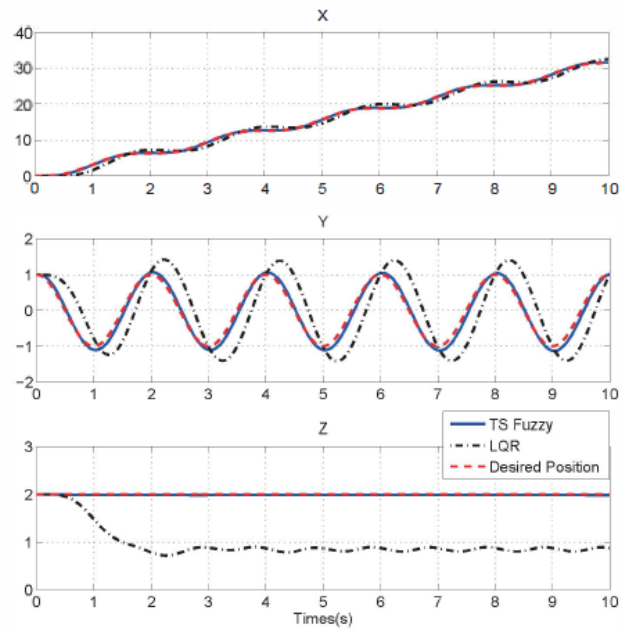
$$[A_i Q + Q A_i^T + B_i Y_i + Y_i^T B_i^T] + [A_j Q + Q A_j^T + B_j Y_j + Y_j^T B_j^T] < 0, (i < j \leq r)$$

$$\begin{bmatrix} -sQ & qQ + Q A_i^T + Y_i^T B_i^T \\ qQ + A_i Q + B_i Y_i & -sQ \end{bmatrix} < 0 \quad (2.6)$$

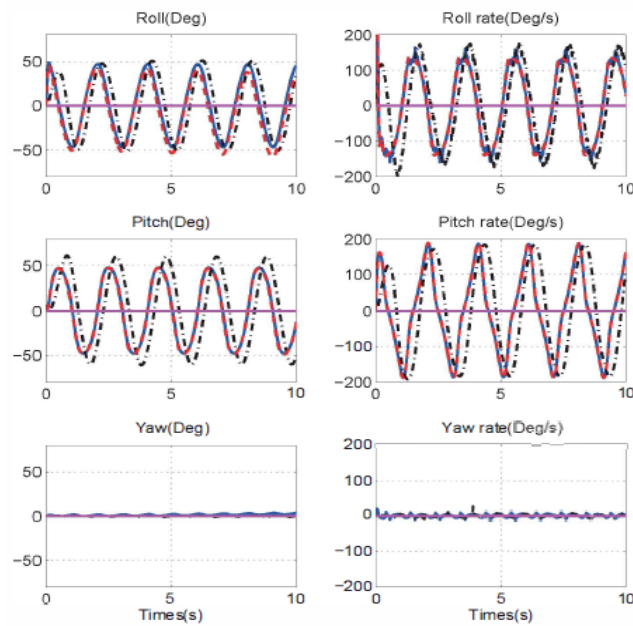
dengan

$$K_i = Y_i Q^{-1}$$

Hasil simulasi sebagaimana pada Gambar 2.8 – 2.9 menunjukkan performa *tracking* yang dirancang dalam paper ini dibandingkan dengan kontroler LQR. Respon *tracking* pada hasil simulasi menunjukkan performa yang lebih baik dari kontroler LQR karena *error tracking* yang dihasilkan. Namun, simulasi tersebut masih dilakukan dengan kondisi tanpa gangguan. Kontrol *tracking* yang dirancang hanya mampu menjamin kestabilan sistem tanpa memperhitungkan adanya gangguan sehingga kekokohan sistem terhadap gangguan belum dijamin.



Gambar 2.8 Simulasi *tracking* posisi x , y dan z



Gambar 2.9 Simulasi *tracking attitude*

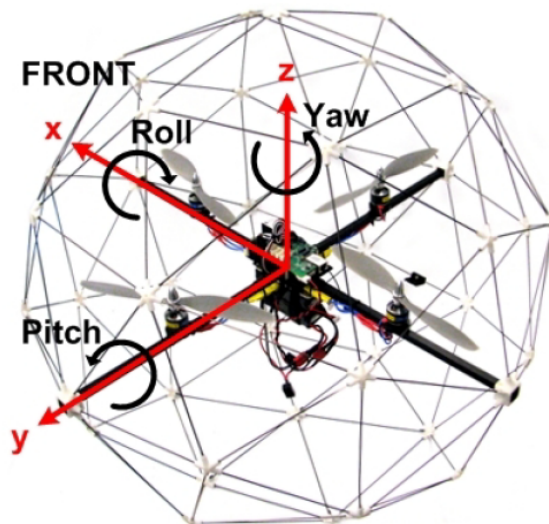
2.2 Teori Dasar

Teori dasar yang digunakan meliputi jenis *quadcopter* yang digunakan beserta pemodelan matematikanya dan teori tentang kontroler yang digunakan pada penelitian ini.

2.2.1 Quanser Qball X4 [20]

Quanser Qball X4 sebagaimana ditunjukkan oleh Gambar 2.12 merupakan *quadcopter* tanpa awak yang menggunakan empat motor dengan pola plus (+). Dua motor saling berlawanan membentuk satu pasang, dimana pasangan pertama (motor 1 dan motor 3) arah pergerakannya diatur searah dengan sumbu x dan berotasi searah dengan jarum jam sedangkan pasangan kedua (motor 2 & motor 4) arah pergerakannya diatur searah dengan sumbu y dan berotasi berlawanan dengan jarum jam. Gaya angkat yang dihasilkan sebanding dengan kecepatan keempat motor

Kelebihan *quadcopter* adalah dapat melakukan *Vertical Take-Off Landing* (VTOL), konstruksi yang lebih sederhana dibandingkan dengan *helicopter* konvensional, mampu terbang *hover*, energi kinetik yang dihasilkan lebih kecil dibandingkan dengan *helicopter* konvensional sehingga lebih aman jika terjadi kecelakaan, dan memiliki kemampuan manuver yang lebih baik dibandingkan dengan UAV bertipe *fixed wing*. Sedangkan kekurangannya adalah memiliki sifat konsumsi energi yang relatif lebih besar jika dibandingkan dengan UAV tipe *fixed wing* dan kontrol kestabilan dari *quadcopter* relatif lebih sulit dicapai karena sifat dari dinamika sistem yang non-linear.



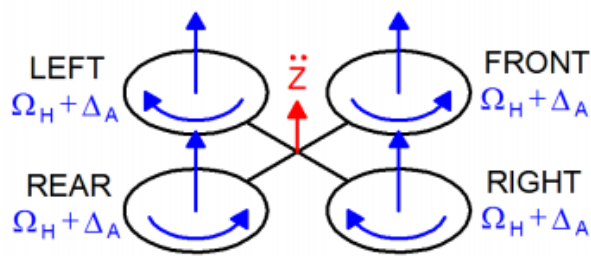
Gambar 2.10 Quanser Qball X4

2.2.2 Gerakan Dasar Quadcopter [17]

Kombinasi kecepatan motor pada *quadcopter* menghasilkan beberapa gerakan dasar. Gerakan ini menjadi dasar gerakan lain pada *quadcopter*.

- *Throttle* atau *height*(Gambar 2.11)

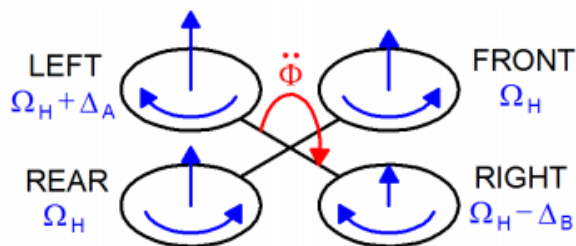
Gerak ini dapat dilakukan dengan menaikkan atau menurunkan kecepatan semua *propeller* dalam jumlah yang sama.



Gambar 2.11 Gerakan *Throttle*

- *Roll* (Gambar 2.12)

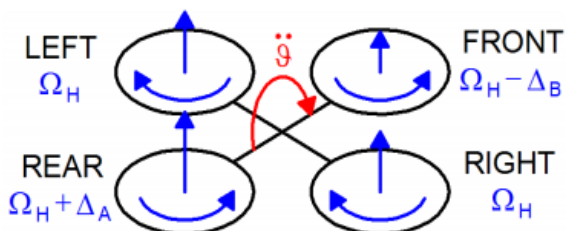
Gerak ini dapat dilakukan dengan menambah atau mengurangi kecepatan salah satu *propeller* yang kiri atau yang kanan.



Gambar 2.12 Gerakan *Roll*

- *Pitch* (Gambar 2.13)

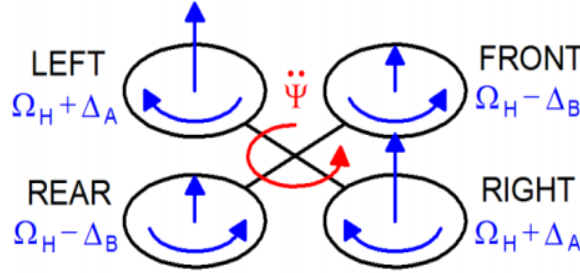
Gerak ini dapat dilakukan dengan menambah atau mengurangi kecepatan salah satu *propeller*, yang depan atau yang belakang.



Gambar 2.13 Gerakan *Pitch*

- *Yaw* (Gambar 2.14)

Gerak ini dapat dilakukan dengan menambah atau mengurangi *propeller* depan belakang dan kanan-kiri secara bersama.



Gambar 2.14 Gerakan *Yaw*

2.2.3 Gaya dan Momen Quadcopter

Quadcopter dianggap tidak memiliki bentuk aerodinamis sehingga gaya aerodinamis dan momen aerodinamis yang terjadi dapat diabaikan. Dari gaya-gaya yang terjadi pada tiap motor di *quadcopter*, dapat dihitung torsi yang terjadi pada *roll*, *pitch*, dan *yaw* dengan Persamaan (2.7) - (2.10)

$$U_1 = F_{T1} + F_{T2} + F_{T3} + F_{T4} \quad (2.7)$$

$$U_2 = F_{T2} - F_{T4} \quad (2.8)$$

$$U_3 = F_{T1} - F_{T3} \quad (2.9)$$

$$U_4 = F_{T1} + F_{T3} - F_{T2} - F_{T4} \quad (2.10)$$

Dimana F_{Ti} adalah gaya angkat (*thrust*) yang dihasilkan oleh setiap rotor yang dinyatakan dalam persamaan (2.11).

$$F_{Ti} = K \frac{\omega}{s + \omega} u_i \quad (2.11)$$

dengan K adalah konstanta gaya dorong, ω adalah lebar bandwidth motor dan u_i adalah sinyal kontrol dari PWM ke- i .

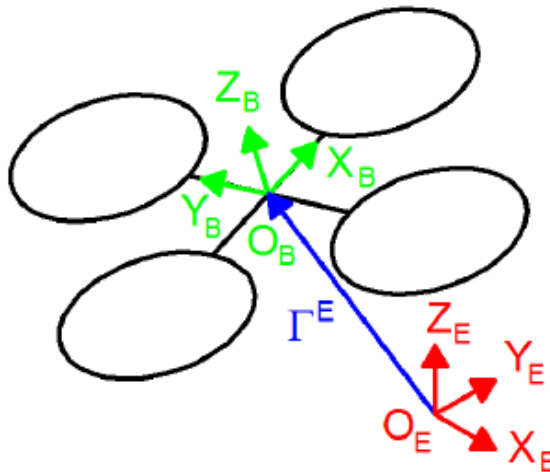
2.2.4 Kinematika Quadcopter [17]

Kinematika adalah cabang dari mekanika klasik yang mempelajari gerakan sebuah benda atau sistem benda tanpa mempersoalkan gaya penyebab gerakan. Untuk menjelaskan gerakan dari quadcopter dengan 6 derajat kebebasan, digunakan 2 sumbu referensi, yaitu:

- Sumbu bumi (*E-frame*) yang dinotasikan dengan X_E , Y_E , dan Z_E ; dan
- Sumbu kerangka quadcopter (*B-frame*) yang dinotasikan dengan X_B , Y_B , dan Z_B .

Posisi linier Γ^E quadcopter sebagaimana Persamaan 2.12 ditentukan dari koordinat vektor *B-frame* asal dan *E-frame* asal terhadap *E-frame* sebagaimana ditunjukkan di Gambar 2.15.

$$\Gamma^E = [X \ Y \ Y]^T \quad (2.12)$$



Gambar 2.15 *E-frame* dan *B-frame*

Posisi sudut Θ^E sebagaimana Persamaan 2.13 quadcopter ditentukan dari sudut *B-frame* terhadap *E-frame*.

$$\Theta^E = [\varphi \ \theta \ \psi]^T \quad (2.13)$$

Matriks rotasi R_θ diperoleh dari kombinasi orientasi *B-frame* terhadap masing-masing sumbu *E-frame* sebagai berikut:

- Matriks rotasi terhadap sumbu X

Dari Gambar 2.16 diperoleh matriks rotasi terhadap sumbu X dengan sudut *roll*(ϕ) sebagaimana Persamaan 2.14.

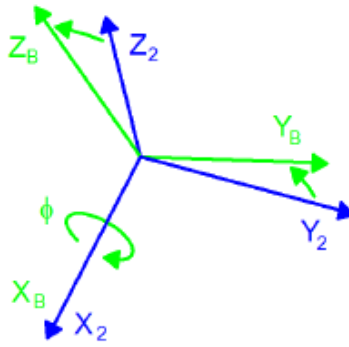
$$\begin{aligned}
X_E &= X_B \cos 0 + Y_B \cos 90 + Z_B \cos 90 \\
Y_E &= X_B \cos 90 + Y_B \cos 0 + Z_B \cos(90 + \phi) \\
Z_E &= X_B \cos 90 + Y_B \sin \phi + Z_B \cos 0
\end{aligned}$$

atau dalam bentuk matriks:

$$\begin{bmatrix} X_E \\ Y_E \\ Z_E \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} X_B \\ Y_B \\ Z_B \end{bmatrix}$$

dengan

$$R(\phi, x) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix} \quad (2.14)$$



Gambar 2.16 Rotasi *B-frame* terhadap sumbu *X* pada *E-frame*

- Matriks rotasi terhadap sumbu *Y*

Dari Gambar 2.19 diperoleh matriks rotasi terhadap sumbu *Y* dengan sudut *pitch*(θ) sebagaimana Persamaan 2.15.

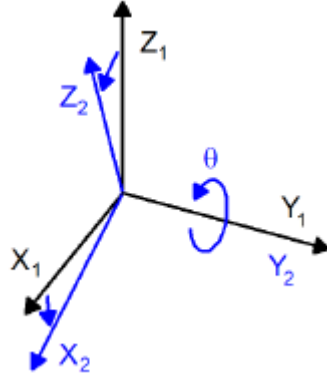
$$\begin{aligned}
X_E &= X_B \cos \theta + Y_B \cos 90 + Z_B \sin \theta \\
Y_E &= X_B \cos 90 + Y_B \cos 0 + Z_B \cos 90 \\
Z_E &= X_B \cos(90 + \theta) + Y_B \cos 90 + Z_B \cos \theta
\end{aligned}$$

atau dalam bentuk matriks:

$$\begin{bmatrix} X_E \\ Y_E \\ Z_E \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix} \begin{bmatrix} X_B \\ Y_B \\ Z_B \end{bmatrix}$$

dengan

$$R(\theta, y) = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix} \quad (2.15)$$



Gambar 2.17 Rotasi *B-frame* terhadap sumbu *Y* pada *E-frame*

- Matriks rotasi terhadap sumbu *Z*

Dari Gambar 2.18 diperoleh matriks rotasi terhadap sumbu *Z* dengan sudut *yaw* (ψ) sebagaimana Persamaan 2.16.

$$X_E = X_B \cos \psi + Y_B \cos(90 + \psi) + Z_B \cos 90$$

$$Y_E = X_B \sin \psi + Y_B \cos \psi + Z_B \cos 90$$

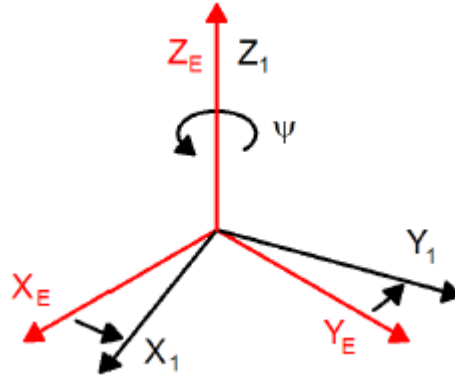
$$Z_E = X_B \cos 90 + Y_B \cos 90 + Z_B \cos 0$$

atau dalam bentuk matriks:

$$\begin{bmatrix} X_E \\ Y_E \\ Z_E \end{bmatrix} = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_B \\ Y_B \\ Z_B \end{bmatrix}$$

dengan

$$R(\psi, z) = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.16)$$



Gambar 2.18 Rotasi *B-frame* terhadap sumbu *Z* pada *E-frame*

Dari Persamaan (2.14), (2.15) dan (2.16) diperoleh matriks rotasi R_{Θ} sebagai berikut:

$$\begin{aligned}
 R_{\Theta} &= R(\psi, z) \times R(\theta, y) \times R(\phi, x) \\
 &= \begin{bmatrix} C\theta C\psi & S\phi S\theta C\psi - C\phi S\psi & C\phi S\theta C\psi + S\phi S\psi \\ C\theta S\psi & S\phi S\theta S\psi + C\phi C\psi & C\phi S\theta S\psi - S\phi C\psi \\ -S\theta & S\phi C\theta & C\phi C\theta \end{bmatrix} \quad (2.17)
 \end{aligned}$$

2.2.5 Dinamika Quadcopter [17]

Jenis pesawat quadcopter dipengaruhi oleh beberapa efek fisik ketika melakukan penerbangan. Efek aerodinamis akibat rotasi *propeller* dan *blade flapping* dapat mempengaruhi sistem ini. Perubahan kecepatan putaran pada *propeller* akan menyebabkan *inertial counter torques*. Ketika quadcopter berputar atau bergerak maju, terjadi efek *gyroscopic* dan gesekan yang mempengaruhi seluruh sistem. Oleh karena itu, pemodelan dinamika dari sistem ini terbilang kompleks.

Model dinamis pada quadcopter dinyatakan dengan beberapa asumsi penyederhanaan. Ketika quadcopter bergerak secara perlahan, efek dari momentum badan quadcopter pada gerakan translasi dapat diabaikan. *Frame* yang digunakan pada quadcopter diasumsikan rigid dan simetris.

Dinamika gerak translasi diperoleh berdasarkan hukum Newton II, yaitu:

$$\begin{aligned}
 \sum F &= m\ddot{\Gamma} \\
 R_{\Theta}U_T + F_g z_g &= m\ddot{\Gamma} \quad (2.18)
 \end{aligned}$$

dengan m adalah massa quadcopter, U_T adalah vector gaya translasi yang dinyatakan sebagaimana Persamaan 2.19 dan F_g adalah gaya gravitasi yang dinyatakan sebagai Persamaan 2.20.

$$U_T = \begin{bmatrix} 0 & 0 & U_1 \end{bmatrix}^T \quad (2.19)$$

$$F_g = -mg \quad (2.20)$$

$$z_g = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$$

Dengan mensubstitusi Persamaan 2.17, 2.19 dan 2.20 ke dalam Persamaan 2.18, diperoleh persamaan dinamika gerak translasi quadcopter sebagai berikut:

$$\begin{bmatrix} C\theta C\psi & S\phi S\theta C\psi - C\phi S\psi & C\phi S\theta C\psi + S\phi S\psi \\ C\theta C\psi & S\phi S\theta S\psi + C\phi C\psi & C\phi S\theta S\psi - S\phi C\psi \\ -S\theta & S\phi C\theta & C\phi C\theta \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ U_1 \end{bmatrix} - mg \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = m \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix}$$

$$\ddot{x} = (\sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi) \frac{U_1}{m} \quad (2.21)$$

$$\ddot{y} = (-\sin \phi \cos \psi + \cos \phi \sin \theta \sin \psi) \frac{U_1}{m} \quad (2.22)$$

$$\ddot{z} = -g + (\cos \phi \cos \theta) \frac{U_1}{m} \quad (2.23)$$

Persamaan gerak rotasi diperoleh dengan persamaan gerak Euler sebagai berikut:

$$\sum \tau = J\ddot{\Theta}$$

$$U_R - C_B(v)\dot{\Theta} = J\ddot{\Theta} \quad (2.24)$$

di mana J adalah matriks inersia sistem, U_R adalah vektor gaya rotasi dan $C_B(v)$ adalah matriks Coriolis-centripetal.

$$\ddot{\Theta} = \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix}$$

$$J = \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix}$$

$$U_R = \begin{bmatrix} U_2 \\ U_3 \\ U_4 \end{bmatrix}$$

$$C_B(v) = \begin{bmatrix} 0 & J_{zz}\psi & -J_{yy}\dot{\theta} \\ -J_{zz}\psi & 0 & J_{xx}\dot{\phi} \\ J_{yy}\dot{\theta} & -J_{xx}\dot{\phi} & 0 \end{bmatrix}$$

sehingga Persamaan 2.24 dapat dituliskan menjadi:

$$\begin{bmatrix} U_2 \\ U_3 \\ U_4 \end{bmatrix} - \begin{bmatrix} 0 & J_{zz}\psi & -J_{yy}\dot{\theta} \\ -J_{zz}\psi & 0 & J_{xx}\dot{\phi} \\ J_{yy}\dot{\theta} & -J_{xx}\dot{\phi} & 0 \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix} \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix}$$

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix}^{-1} \left(\begin{bmatrix} U_2 \\ U_3 \\ U_4 \end{bmatrix} - \begin{bmatrix} 0 & J_{zz}\psi & -J_{yy}\dot{\theta} \\ -J_{zz}\psi & 0 & J_{xx}\dot{\phi} \\ J_{yy}\dot{\theta} & -J_{xx}\dot{\phi} & 0 \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \right) \quad (2.25)$$

Persamaan dinamika rotasi untuk masing-masing gerak roll, pitch dan yaw sebagaimana dituliskan pada Persamaan (2.26) – (2.28).

$$\ddot{\phi} = \frac{U_2 l}{J_{xx}} - \frac{\dot{\theta} \psi}{J_{xx}} (J_{zz} - J_{yy}) \quad (2.26)$$

$$\ddot{\theta} = \frac{U_3 l}{J_{yy}} - \frac{\dot{\phi} \psi}{J_{yy}} (J_{xx} - J_{zz}) \quad (2.27)$$

$$\ddot{\psi} = \frac{U_4}{J_{zz}} - \frac{\dot{\phi} \dot{\theta}}{J_{zz}} (J_{yy} - J_{xx}) \quad (2.28)$$

Di mana J_{xx} merupakan momen inersia quadcopter terhadap sumbu x, J_{yy} merupakan momen inersia quadcopter terhadap sumbu y dan J_{zz} merupakan momen inersia quadcopter terhadap sumbu z. Nilai parameter quadcopter yang akan digunakan pada penelitian ini sebagaimana pada Tabel 2.1.

Tabel 2.1Paramater quadcopter [20]

Simbol	Parameter	Nilai	Unit
K	Input PWM dari aktuator	120	N
l	Jarak baling-baling dan pusat quadcopter	0,2	m
J_{xx}	Momen Inersia pada sumbu x	0,03	kg.m ²
J_{yy}	Momen Inersia pada sumbu y	0,03	kg.m ²
J_{zz}	Momen Inersia pada sumbu z	0,04	kg.m ²
m	Massa quadcopter	3,5	kg
ω	<i>Bandwidth</i> quadcopter	15	rad/sec

.0

2.2.6 Metode *Feedback Linearization*[21]

Feedbacklinearization adalah suatu metode yang digunakan untuk menghilangkan bagian nonlinier dari suatu sistem. Dalam hal ini *feedbacklinearization* yang dimaksud adalah *input-statefeedbacklinearization*. Pada sistem MIMO,*input-statefeedbacklinearization* dapat digunakan untuk menghilangkan interaksi di dalam sistem tersebut sehingga sistem MIMO seolah-olah bekerja seperti sistem *Single Input Single Output* (SISO). Hal ini dapat dilakukan dengan mendefinisikan input baru sedemikian rupa sehingga dapat menghilangkan nonlinieritas sistem dan interaksi di dalam sistem. Metode seperti ini sering disebut juga kontrol *decoupling*.

Sebuah sistem dinyatakan dalam bentuk persamaan *statespace* pada Persamaan 2.29.

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= k_1 x_1 x_2 + k_2 u\end{aligned}\tag{2.29}$$

Untuk menghilangkan nonlinieritasi dalam Persamaan 2.26, ditentukan persamaan kontrol pada Persamaan 2.30 sehingga diperoleh sistem linier sebagaimana Persamaan 2.31.

$$u = \frac{1}{k_2} (v - k_1 x_1 x_2)\tag{2.31}$$

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= v\end{aligned}\tag{2.31}$$

Kemudian ditentukan kemudian ditentukan persamaan kontroller *linear state feedback* sebagaimana Persamaan 2.32. Nilai parameter kontroler didasarkan pada model referensi dengan karakteristik respon transient sesuai keinginan.

$$v = -ax_1 - bx_2 + cu^*\tag{2.32}$$

Dengan mensubstitusikan Persamaan (2.32) ke dalam Persamaan (2.30), maka didapatkan persamaan *state feedback* sebagai berikut.

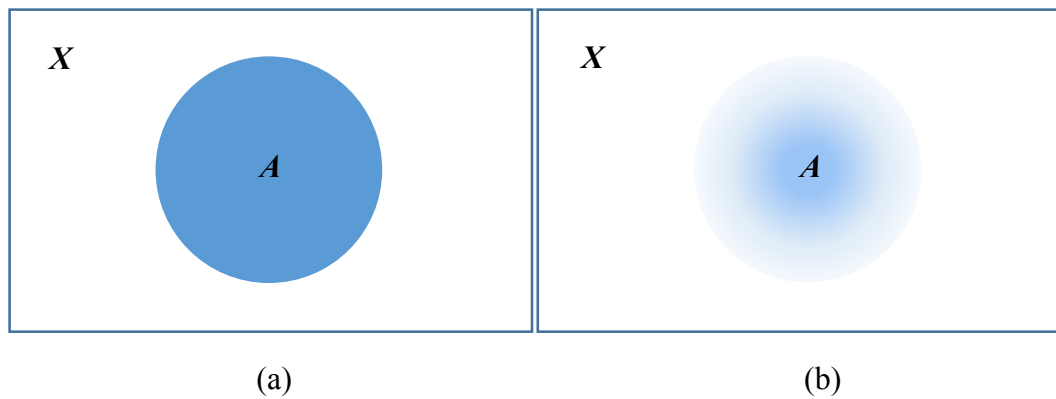
$$u = \frac{1}{k_2}(-ax_1 - bx_2 + cu^* - k_1x_1x_2)\tag{2.33}$$

2.2.7 Logika Fuzzy [22],[23]

Logika fuzzy merupakan salah satu komponen pembentuk *soft computing*. Logika ini pertama kali diperkenalkan oleh Prof. Lotfi A. Zadeh pada tahun 1965. Dasar logika ini adalah teori himpunan fuzzy. Pada teori himpunan fuzzy, peranan derajat keanggotaan sebagai penentu keberadaan elemen dalam suatu himpunan sangat penting. Nilai keanggotaan atau derajat keanggotaan atau *membership function* menjadi ciri utama dari penalaran dengan logika fuzzy ini. Berikut penjelasan mengenai komponen penting dalam logika fuzzy yang meliputi himpunan fuzzy, fungsi keanggotaan, operator himpunan dan sistem inferensi fuzzy.

2.2.7.1 Himpunan Fuzzy

Himpunan fuzzy muncul setelah berkembangnya himpunan klasik. Misalkan X adalah sebuah ruang yang didalamnya terdapat elemen-elemen yang dilambangkan dengan x , maka $X = \{x\}$. Himpunan fuzzy A dalam X ditandai dengan fungsi keanggotaan $\mu_A(x)$ dalam interval $[0, 1]$, dengan nilai $\mu_A(x)$ pada x merepresentasikan derajat keanggotaan x dalam A . Dengan demikian, semakin dekat nilai $\mu_A(x)$ dengan 1, semakin tinggi erajat keanggotaan x dalam A . Hal ini yang membedakan himpunan fuzzy dengan himpunan klasik yang hanya memiliki nilai benar atau salah secara tegas sebagaimana ilustrasi yang ditunjukkan pada Gambar 2.19.



Gambar 2.19 (a) himpunan klasik, (b) himpunan fuzzy.

2.2.7.2 Fungsi Kenggotaan

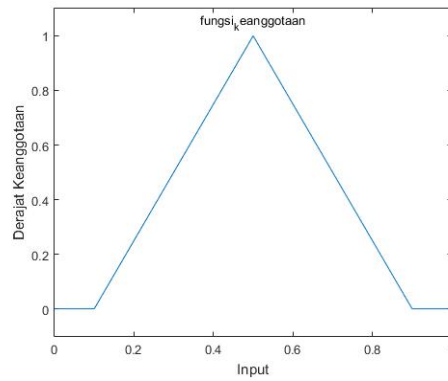
Fungsi keanggotaan adalah suatu kurva yang menunjukkan pemetaan titik-titik input data ke dalam nilai keanggotaannya yang memiliki interval antara 0 sampai dengan 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah melalui pendekatan fungsi. Ada beberapa fungsi yang dapat digunakan, antara lain:

a. Fungsi segitiga

Fungsi segitiga pada dasarnya merupakan gabungan antara 2 garis linier seperti terlihat pada Gambar 2.20 dengan persamaan matematika sebagaimana Persamaan 2.34.

$$\mu[x] = \begin{cases} 0; & x \leq a \text{ atau } x \geq c \\ \frac{x-a}{b-a}; & a \leq x \leq b \\ \frac{b-x}{c-b}; & b \leq x \leq c \end{cases} \quad (2.34)$$

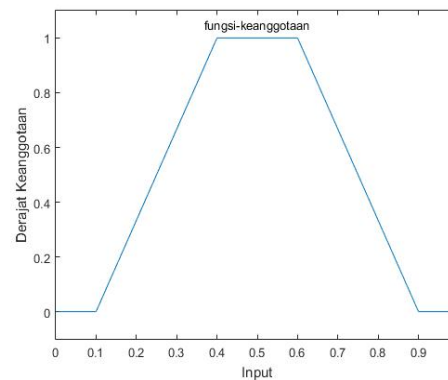
dengan a , b dan c adalah parameter yang menyatakan letak koordinat ketiga pada sumbu x .



Gambar 2.20 Fungsi segitiga

b. Fungsi trapesium

Fungsi trapesium ditentukan oleh empat parameter a , b , c dan d yang menyatakan letak koordinat sudut trapesium pada sumbu x sebagaimana ditunjukkan di Gambar 2.21 dan dinyatakan pada Persamaan 2.35.



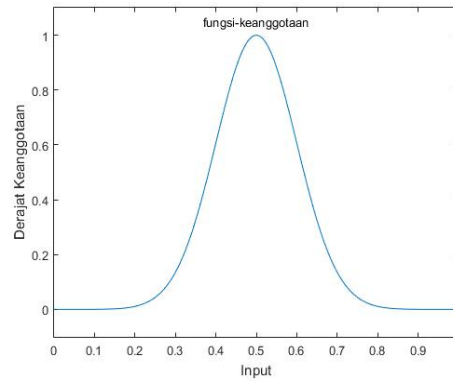
Gambar 2.21 Fungsi trapesium

$$\mu[x] = \begin{cases} 0; & x \leq a \text{ atau } x \geq c \\ \frac{x-a}{b-a}; & a \leq x \leq b \\ 1; & b \leq x \leq c \\ \frac{d-x}{d-c}; & c \leq x \leq d \end{cases} \quad (2.35)$$

c. Fungsi Gaussian

Fungsi Gaussian ditentukan oleh dua parameter yaitu parameter c yang menentukan titik pusat dan parameter σ yang menentukan lebar fungsi sebagaimana dinyatakan pada Persamaan 2.36 dan ditunjukkan di Gambar 2.22.

$$\mu[x] = \exp\left[-\frac{1}{2}\left(\frac{x-c}{\sigma}\right)^2\right] \quad (2.36)$$

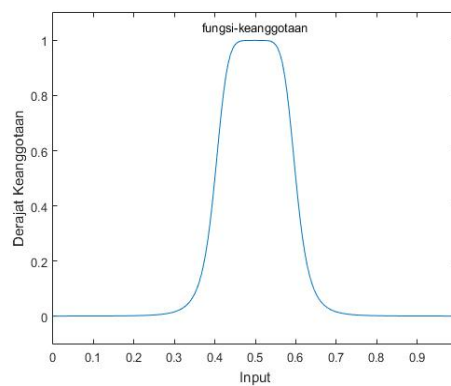


Gambar 2.22 Fungsi Gaussian

d. Fungsi *Generalized Bell*

Fungsi *Generalized Bell* ditentukan oleh tiga parameter, yaitu parameter a dan b yang menentukan lebar fungsi serta parameter c yang menentukan titik pusat fungsi. Parameter b harus bernilai positif. Fungsi ini dinyatakan dengan Persamaan 2.37 dan ditunjukkan pada Gambar 2.23

$$\mu[x] = \frac{1}{1 + \left|\frac{x-c}{a}\right|^{2b}} \quad (2.37)$$



Gambar 2.25 Fungsi *Generalized Bell*

2.2.7.3 Operator Himpunan

Ada beberapa operasi himpunan yang digunakan untuk mengkombinasikan dan memodifikasi himpunan fuzzy, antara lain:

1. Operator AND

Operator ini dapat diartikan sebagai operasi interseksi pada himpunan. Derajat keanggotaan sebagai hasil operasi dengan operator AND diperoleh dengan mengambil derajat keanggotaan terkecil antar elemen pada himpunan-himpunan yang bersangkutan. Operator ini dapat dinyatakan sebagaimana Persamaan 2.38.

$$\mu_{A \cap B} = \min(\mu_A(x), \mu_B(y)) \quad (2.38)$$

2. Operator OR

Operator ini dapat diartikan sebagai operasi union pada himpunan. Derajat keanggotaan sebagai hasil dari operasi OR diperoleh dengan mengambil derajat keanggotaan terbesar antar elemen pada himpunan-himpunan yang bersangkutan. Operator ini dapat dinyatakan sebagaimana Persamaan 2.36.

$$\mu_{A \cup B} = \max(\mu_A(x), \mu_B(y)) \quad (2.39)$$

3. Operator NOT

Operator ini dapat diartikan sebagai operasi komplemen pada himpunan. Derajat keanggotaan sebagai hasil dari operasi NOT diperoleh dengan mengurangkan derajat keanggotaan elemen pada himpunan yang bersangkutan dari 1. Operator ini dapat dinyatakan sebagaimana Persamaan 2.40.

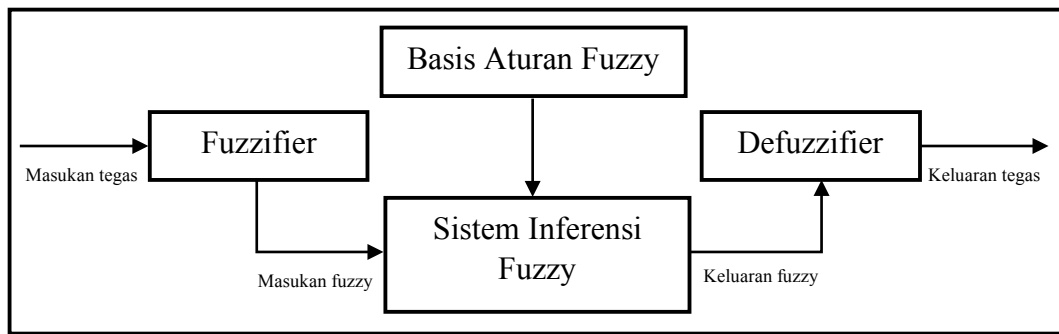
$$\mu_A = 1 - \mu_A(x) \quad (2.40)$$

2.2.7.4 Sistem Inferensi Fuzzy

Bentuk umum dari aturan yang digunakan sebagaimana Persamaan 2.41.

$$\text{If } x \text{ is } A \text{ Then } y \text{ is } B \quad (2.41)$$

dengan x adalah variabel premis, y adalah konsekuen, A dan B adalah himpunan fuzzy.



Gambar 2.24 Struktur umum sistem inferensi fuzzy

Aturan fuzzy yang telah disusun digunakan dalam sistem inferensi fuzzy sehingga dapat diperoleh keputusan dari setiap premis yang dibuat. Struktur umum sistem inferensi fuzzy sebagaimana ditunjukkan pada Gambar 2.24. Dari gambar tersebut dapat dilihat bahwa dalam sistem inferensi fuzzy terdapat empat bagian pokok, antara lain:

- Fuzzifier, berfungsi mentransformasikan sinyal masukan *crisp*/tegas ke himpunan fuzzy.
- Basis aturan fuzzy, berisi data aturan yang berfungsi memetakan nilai masukan fuzzy ke nilai keluaran fuzzy.
- Sistem inferensi fuzzy, merupakan inti dari logika fuzzy yang berfungsi sebagai pengambil keputusan.
- Defuzzifier, berfungsi mentransformasikan keluaran fuzzy menjadi keluaran tegas.

2.2.7.5 Metode defuzzifikasi

Proses defuzzifikasi keluaran fuzzy menjadi keluaran tegas dapat dilakukan dengan beberapa metode sebagai berikut:

1. Metode *center of gravity*

Dengan metode ini, keluaran tegas dapat diperoleh melalui Persamaan 2.42.

$$y = \frac{\sum_{i=1}^r b_i \int_y \mu_i(y) dy}{\sum_{i=1}^r \int_y \mu_i(y) dy} \quad (2.42)$$

di mana y adalah keluaran tegas, r adalah jumlah aturan, b_i adalah titik tengah dari fungsi keanggotaan dari aturan ke- i dan μ_i adalah derajat keanggotaan dari aturan ke- i .

2. Metode *center average*

Dengan metode ini, keluaran tegas dapat diperoleh melalui Persamaan 2.43.

$$y = \frac{\sum_{i=1}^r b_i \mu_i(y)}{\sum_{i=1}^r \mu_i(y)} \quad (2.43)$$

di mana y adalah keluaran tegas, r adalah jumlah aturan, b_i adalah titik tengah dari fungsi keanggotaan dari aturan ke- i dan μ_i adalah derajat keanggotaan dari aturan ke- i .

2.2.8 Model Fuzzy Takagi-Sugeno (T-S) [14],[15]

Sistem fuzzy T-S dikenal sebagai sistem fuzzy fungsional. Berbeda dengan sistem fuzzy Mamdani, konsekuen dalam sistem fuzzy T-S berupa fungsi. Sistem fuzzy T-S dapat digunakan untuk pemodelan sistem nonlinier. Model fuzzy T-S akan merepresentasikan sistem nonlinier ke dalam beberapa model linier untuk daerah kerja yang ditentukan. Model sistem secara keseluruhan merupakan perpaduan antara model-model linier tersebut. Premis dari model fuzzy T-S merupakan titik kerja yang digunakan pada proses linierisasi sehingga aturan dalam model menggambarkan relasi linier antara masukan dan keluaran dari sistem nonlinier.

Model fuzzy T-S dapat dinyatakan dengan Persamaan 2.44.

Aturan plant ke- i :

$$\begin{aligned} \text{If} \quad & z_1 \text{ is } M_{i1} \text{ and } z_2 \text{ is } M_{i2} \dots z_j \text{ is } M_{ij} \\ \text{Then} \quad & \dot{x}(t) = A_i x(t) + B_i u(t) \\ & y(t) = Cx(t) + D_i u(t) \quad ; i = 1, 2, \dots, q \end{aligned} \quad (2.44)$$

dengan q sebagai jumlah aturan fuzzy dan j adalah jumlah himpunan fuzzy dalam satu aturan, M_{ij} sebagai himpunan fuzzy, z merupakan variabel pada bagian premis yang dapat berupa fungsi dari variabel *state* dan u adalah masukan sistem. Vektor

state $x \in \mathbb{R}^n$, vektor keluaran sistem $y \in \mathbb{R}^q$, sedangkan A , B , C , dan D adalah matriks dengan ukuran sesuai.

Operator AND digunakan dalam tahap inferensi pada model fuzzy T-S, sedangkan metode *center average* digunakan dalam tahap defuzzifikasi. Aturan plant dalam model fuzzy T-S secara keseluruhan dapat dituliskan sebagaimana Persamaan 2.45.

$$\begin{aligned}\dot{x}(t) &= \sum_i h_i [A_i x(t) + B_i u(t)] \\ y(t) &= \sum_i h_i [C_i x(t) + D_i u(t)]\end{aligned}\quad (2.45)$$

dengan

$$h_i = \frac{\mu_i(z)}{\sum_i \mu_i(z)}; \mu_i(z) = \prod_{j=1}^p M_{ij}(z_j)$$

Kontroler dapat disusun dengan konsep *Parallel Distributed Compensation* (PDC). Dalam konsep ini, tiap aturan kontroler dirancang berdasarkan aturan model plant linier yang bersesuaian dengan himpunan fuzzy yang sama. Untuk setiap aturan, dapat digunakan teknik kontrol linier. Hasil keseluruhan kontroler fuzzy secara umum merupakan perpaduan dari hasil setiap kontroler linier. Kontroler fuzzy dengan konsep PDC dapat dituliskan sebagaimana Persamaan 2.46.

Aturan kontroler ke- i :

$$\begin{aligned}\text{If} \quad & z_1 \text{ is } M_{i1} \text{ and } \dots z_j \text{ is } M_{ij} \\ \text{Then} \quad & u(t) = K_i(t) e(t) \quad ; i = 1, 2, \dots, r\end{aligned}\quad (2.46)$$

dengan

$$e(t) = r(t) - y(t)$$

$e(t)$ adalah nilai kesalahan *tracking* yang didapat dari selisih antara referensi yang diberikan dan keluaran. K_i adalah nilai penguat untuk antuan kontroler ke- i . Oleh karena itu, sinyal kontrol dari kontroler fuzzy dapat dituliskan sebagaimana Persamaan 2.47.

$$u(t) = \sum_{i=1}^r m_i(z) [K_i e(t)] \quad (2.47)$$

2.2.9 Analisis Kestabilan Lyapunov [24]

Stabilitas merupakan hal yang sangat penting dalam sebuah sistem kontrol. Menurut Lyapunov, kestabilan sistem dapat ditentukan dari energi yang disimpan oleh sistem tersebut. Sistem dikatakan stabil jika energi yang disimpan semakin lama semakin kecil hingga akhirnya habis. Sulit menentukan fungsi matematika dari energi suatu sistem, maka Lyapunov membuat suatu fungsi energi buatan. Fungsi ini bergantung kepada vektor *state* sistem yang dapat dinyatakan dengan $V(x)$ dan turunannya dapat memberikan informasi mengenai kestabilan sistem tanpa harus mendapatkan solusi persamaan diferensial sistem. Analisa kestabilan Lyapunov dapat digunakan di sistem linier dan sistem nonlinier.

Sistem dikatakan stabil asimtotik dalam pemahaman Lyapunov apabila memenuhi apabila memenuhi kondisi pada Persamaan (2.48).

$$\dot{x} = f(x, t); f(0, t) = 0 \quad (2.48)$$

dan jika terdapat fungsi skalar $V(x)$ yang kontinyu dan memenuhi syarat berikut:

1. $V(x)$ definit positif, untuk semua nilai $x \neq 0$
2. $\dot{V}(x)$ definit negatif, untuk semua nilai $x \neq 0$
3. $V(0) = 0$

maka *state* ekuilibrium akan stabil asimtotik pada origin terlihat saat fungsi energi tersebut definit positif, berarti ada energi dalam tersebut. Kemudian, $\dot{V}(x)$ atau laju perubahan $V(x)$ yang definit negatif menyatakan bahwa energi tersebut berangsur-angsur berkurang hingga menyebabkan fungsi $V(x)$ bernilai 0 dan *state* sistem bergerak menuju titik ekuilibrium di origin.

Dalam sebuah sistem linier waktu kontinyu, misalkan:

$$\dot{x} = Ax \quad (2.49)$$

dengan x adalah vektor *state* dan A adalah matriks konstan, maka dipilih fungsi Lyapunov dalam bentuk kuadratik yang memungkinkan, yaitu:

$$V(x) = x^T Px \quad (2.50)$$

P adalah matriks simetris bernilai definit positif agar $V(x) > 0$ untuk semua nilai x , sesuai teorema kestabilan Lyapunov. Sementara turunan dari $V(x)$ adalah sebagai berikut.

$$\begin{aligned}
 \dot{V}(x) &= \dot{x}^T P x + x^T P \dot{x} \\
 &= [Ax]^T P x + x^T P [Ax] \\
 &= x^T A^T P x + x^T P A x \quad (2.51) \\
 &= x^T [A^T P + P A] x \\
 &= -x^T Q x
 \end{aligned}$$

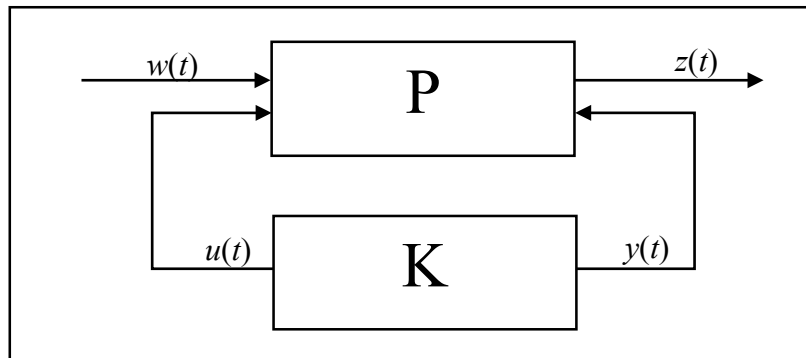
dengan

$$Q = -[A^T P + P A] \quad (2.52)$$

Agar sistem dapat stabil asimtotik, syarat cukup yang harus dipenuhi adalah Q bernilai definit positif.

2.2.10 Performa H^∞ [25],[26]

Salah satu metode kontrol mengenai kekokohan yang sering digunakan adalah kontrol H^∞ . Pada metode kontrol ini, pengaruh gangguan dari luar terhadap performansi sistem akan diminimalkan sesuai level pelemahan yang ditentukan. Bentuk konfigurasi dasar sistem *feedback* yang digunakan dalam metode kontrol ini sebagaimana ditunjukkan pada Gambar 2.25.



Gambar 2.25 Bentuk dasar sistem *feedback* pada kontrol H^∞ .

Terdapat dua masukan serta keluaran dari plant (P). masukan tersebut adalah $u(t)$ dan $w(t)$ yang merupakan sinyal kontrol dan sinyal gangguan yang diberikan pada sistem. Sementara keluaran sistem adalah $z(t)$ dan $y(t)$ yang

merupakan keluaran performansi dan keluaran terukur. K adalah nilai penguat yang dirancang sebagai kontroler.

Untuk mengetahui apakah gangguan dari luar tersebut dapat dilemahkan atau tidak, pengaruh $w(t)$ terhadap $z(t)$ dapat diketahui dengan menghitung ∞ -norm fungsi alih $T_{zw}(s)$ sebagaimana Persamaan 2.50.

$$\|T_{zw}\|_{\infty} = \sup_{\omega} |T_{zw}(j\omega)| = \sup_{\|w(t)\|_2 \neq 0} \frac{\|z(t)\|_2}{\|w(t)\|_2} \quad (2.53)$$

Persamaan 2.50 menyatakan bahwa ∞ -norm fungsi alih $T_{zw}(s)$ merupakan nilai maksimum dari L_2 -gain yaitu perbandingan antara L_2 -norm $z(t)$ dan L_2 -norm $w(t)$. Performansi H_{∞} yang berkaitan dengan nilai kesalahan *tracking* dinyatakan dalam L_2 -gain sebagaimana Persamaan 2.54.

$$\frac{\|z(t)\|_2}{\|w(t)\|_2} = \frac{\sqrt{\int_0^{tf} \{[x(t) - x_r(t)]^T Q [x(t) - x_r(t)]\} dt}}{\sqrt{\int_0^{tf} \bar{w}(t)^T \bar{w}(t) dt}} < \rho \quad (2.54)$$

atau

$$\int_0^{tf} \{[x(t) - x_r(t)]^T Q [x(t) - x_r(t)]\} dt \leq \rho^2 \int_0^{tf} \bar{w}(t)^T \bar{w}(t) dt$$

dengan $\bar{w}(t)$ merupakan matriks augmentasi dari $w(t)$ atau gangguan dan $x_r(t)$ atau sinyal referensi, Q merupakan matriks pembobot definit positif.

Persamaan 2.54 memiliki makna bahwa segala efek yang diberikan oleh $\bar{w}(t)$ di nilai kesalahan *tracking* bisa dilemahkan sampai kurang dari level pelemahan ρ yang ditentukan. Apapun nilai $\bar{w}(t)$ yang ada, nilai norm dari gain $\bar{w}(t)$ ke $x(t) - x_r(t)$ harus sama dengan atau kurang dari nilai ρ^2 .

2.2.11 Linear Matrix Inequality (LMI) [19]

Bentuk umum LMI

$$F(x) = F_0 + \sum_{i=1}^m x_i F_i > 0$$

dengan x merupakan variabel yang berbentuk matriks dan F_i merupakan matriks simetris. Pertidaksamaan tersebut memiliki arti bahwa fungsi $F(x)$ bernilai definit positif. Selain itu, pertidaksamaan memiliki pembatas konveks pada x . karakteristik lain dari LMI jika terdapat beberapa bentuk LMI

$$F_1(x) > 0, F_2(x) > 0, \dots, F_n(x) > 0$$

maka dapat dinyatakan ke dalam bentuk LMI tunggal

$$\text{diag}(F_1(x) > 0, F_2(x) > 0, \dots, F_n(x) > 0)$$

Teorema Lyapunov merupakan salah satu contoh bentuk dari LMI. Persamaan (2.47) dan (2.48) dapat dinyatakan ke dalam bentuk LMI sebagai berikut

$$V(x) = x^T P x > 0$$

$$\dot{V}(x) = x^T [A^T P + P A] x < 0$$

$$\dot{V}(x) = x^T Q x < 0$$

sehingga secara keseluruhan analisis kestabilan Lyapunov sistem dapat dinyatakan sebagai

$$P > 0$$

$$A^T P + P A < 0$$

Jika kendala merupakan pertidaksamaan yang nonlinier, maka kendala tersebut dapat diubah menjadi LMI dengan menggunakan metode *Schur Complement*. Contoh:

$$L(x) < 0 \rightarrow \begin{cases} E(x) < 0 \\ H(x) - G(x)E(x)^{-1}F(x) < 0 \end{cases}$$

dengan

$$E(x) = E(x)^T, H(x) = H(x)^T \text{ dan } G(x) = F(x)^T$$

dapat disederhanakan menjadi bentuk LMI sebagai berikut.

$$L(x) = \begin{bmatrix} E(x) & F(x) \\ G(x) & H(x) \end{bmatrix} < 0$$

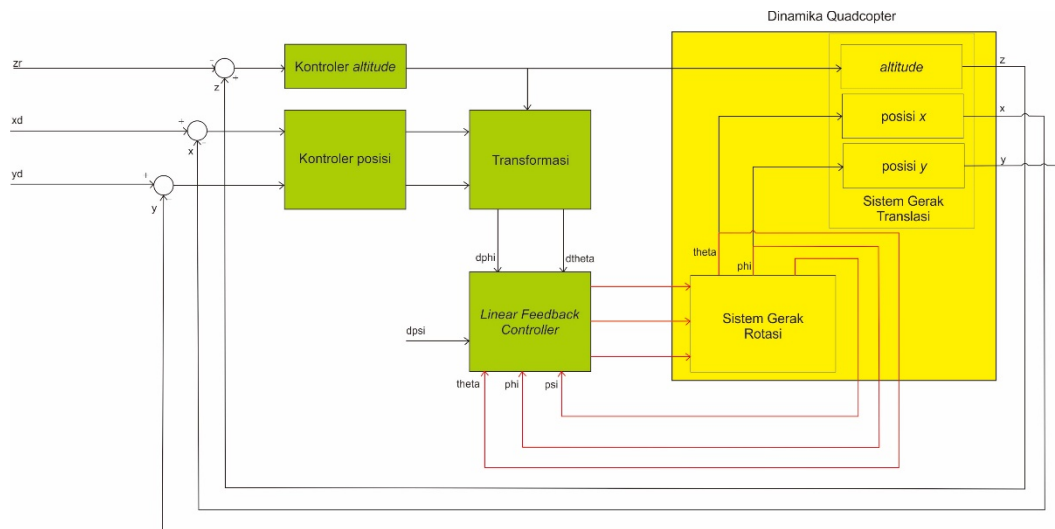
BAB 3

METODOLOGI PENELITIAN

Dalam penelitian ini, sistem quadcopter dibagi menjadi dua yaitu sistem gerak rotasi sebagai *inner-loop* dan sistem gerak translasi sebagai *outer-loop*. Sistem kontrol dirancang dengan metode yang berbeda untuk masing-masing sistem gerak tersebut. Pada sistem gerak rotasi, kontroler dirancang menggunakan metode *input-state feedback linearization*. Sedangkan pada sistem gerak translasi, dilakukan pemecahan menjadi subsistem *altitude* dan subsistem posisi. Pada subsistem *altitude*, kontroler dirancang menggunakan metode fuzzy T-S dan pada subsistem posisi, kontroler dirancang menggunakan metode Proporsional dan Derivatif (PD). Diagram blok sistem kontrol pada quadcopter dapat dilihat pada Gambar 3.1.

Sinyal input gerak *roll* dan *pitch* (ϕ_r dan θ_r) berasal dari keluaran kontroler PD pada subsistem posisi (U_x dan U_y), sehingga nilai ϕ_r dan θ_r akan disesuaikan dengan posisi x dan y yang diinginkan. Sedangkan sinyal input untuk gerak *yaw* (ψ_r) adalah 0 rad. Dengan persamaan *input-state feedback* yang dirancang, akan dihasilkan sinyal kontrol U_2 , U_3 dan U_4 sehingga sudut *roll* dan *pitch* dapat mencapai nilai referensi yang diberikan dan *yaw* dapat mempertahankan nilai referensi yang diberikan. Selanjutnya, keluaran sistem gerak rotasi akan digunakan dalam sistem gerak translasi.

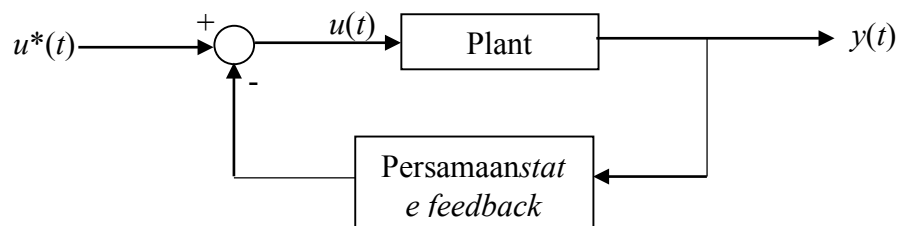
Untuk subsistem *altitude*, sinyal referensi yang diberikan adalah sinyal pulsa dengan *offset* 0 dan amplitudo 5. *Error tracking* diperoleh dari selisih antara nilai state terukur dan referensi. Sinyal kontrol U_1 merupakan hasil perkalian antara *error tracking* dengan penguat K_1 pada kontroler fuzzy. Sedangkan untuk subsistem posisi digunakan kontroler PD dengan sinyal input x_d dan y_d adalah 0.



Gambar 3.1 Diagram blok sistem kontrol pada quadcopter

3.1 Sistem Gerak Rotasi

Pada sistem ini, kontroler dirancang menggunakan metode *input-state feedback linearization* dengan diagram blok sebagaimana pada Gambar 3.2. Sistem gerak rotasi akan di-*decouple* sehingga interaksi antar komponen persamaan gerak *roll*, *pitch* dan *yaw* dapat dieliminasi. Kontroler *linear state feedback* dirancang dengan pendekatan model referensi berdasarkan penempatan *pole* dominan sesuai karakteristik respon transien yang diinginkan. Persamaan *input-state feedback* diperoleh dengan mensubstitusi persamaan model referensi ke dalam persamaan model sistem gerak rotasi.



Gambar 3.2 Diagram blok sistem gerak rotasi

3.1.1 Model Matematika Sistem Gerak Rotasi

Model matematika sistem gerak rotasi sebagaimana pada Persamaan 2.26 – 2.28 dapat dituliskan sebagai berikut:

$$\begin{aligned}
\ddot{\phi} &= \frac{U_2 l}{J_{xx}} - \frac{qr}{J_{xx}} (J_{zz} - J_{yy}) \\
\ddot{\theta} &= \frac{U_3 l}{J_{yy}} - \frac{pr}{J_{yy}} (J_{xx} - J_{zz}) \\
\ddot{\psi} &= \frac{U_4}{J_{zz}} - \frac{pq}{J_{zz}} (J_{yy} - J_{xx})
\end{aligned} \tag{3.1}$$

Dalam bentuk persamaan state-space, Persamaan (3.1) dapat dinyatakan sebagai:

$$\begin{aligned}
\dot{x}_7 &= x_8 \\
\dot{x}_8 &= \frac{U_2 l}{J_{xx}} - \frac{J_{zz} - J_{yy}}{J_{xx}} x_{10} x_{12} \\
\dot{x}_9 &= x_{10} \\
\dot{x}_{10} &= \frac{U_3 l}{J_{yy}} - \frac{J_{xx} - J_{zz}}{J_{yy}} x_8 x_{12} \\
\dot{x}_{11} &= x_{12} \\
\dot{x}_{12} &= \frac{U_4}{J_{zz}} - \frac{J_{yy} - J_{xx}}{J_{zz}} x_8 x_{10}
\end{aligned} \tag{3.2}$$

atau

$$\begin{bmatrix} \dot{x}_7 \\ \dot{x}_8 \\ \dot{x}_9 \\ \dot{x}_{10} \\ \dot{x}_{11} \\ \dot{x}_{12} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_7 \\ x_8 \\ x_9 \\ x_{10} \\ x_{11} \\ x_{12} \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{J_{zz} - J_{yy}}{J_{xx}} x_{10} x_{12} \\ 0 \\ -\frac{J_{xx} - J_{zz}}{J_{yy}} x_{10} x_{12} \\ 0 \\ -\frac{J_{yy} - J_{xx}}{J_{zz}} x_{10} x_{12} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{l}{J_{xx}} U_2 \\ 0 \\ \frac{l}{J_{yy}} U_3 \\ 0 \\ \frac{1}{J_{zz}} U_4 \end{bmatrix}$$

di mana

$$\begin{aligned}
x_7 &= \phi \\
x_8 &= \dot{\phi} \\
x_9 &= \theta \\
x_{10} &= \dot{\theta} \\
x_{11} &= \psi \\
x_{12} &= \dot{\psi}
\end{aligned}$$

3.1.2 Kontroler *Linear State Feedback*

Kontroler dirancang dengan pendekatan model referensi berdasarkan karakteristik respon transien yang diinginkan, yaitu:

- a) *Zero overshoot*;
- b) *zero offset*; dan
- c) *settling time* (t_s) 1 detik $\pm 2\%$.

Dalam hal ini, respon transien yang dimaksud adalah respon transien sistem lup tertutup terhadap sinyal step.

Persamaan gerak roll, pitch dan yaw adalah persamaan orde 2. Untuk itu, model referensi dirancang dengan jumlah orde yang sama sebagaimana Persamaan (3.3).

$$\frac{C(s)}{U(s)} = \frac{(a \times b)}{(s^2 + (a + b)s + (a \times b))} \quad (3.3)$$

Sistem lup tertutup orde 2 yang mempunyai rasio peredaman $\zeta > 1$ dengan 2 pole negatif riil:

$$\text{pole I: } s_1 = -a \quad \text{pole II: } s_2 = -b$$

di mana

$$a \neq b$$

jika s_1 terletak lebih dekat dengan sumbu imajiner dibandingkan s_2 atau $a \ll b$, maka s_1 adalah pole dominan. Pole dominan adalah pole yang paling berpengaruh terhadap karakteristik respon transien sistem lup tertutup tersebut. Semakin besar nilai $|s_2|$ maka pengaruhnya terhadap karakteristik respon transien sistem lup tertutup tersebut sangat kecil atau dapat diabaikan.

Jika $U(s) = 1/s$ dimasukkan pada Persamaan (3.3), maka diperoleh:

$$C(s) = \frac{1}{s} + \frac{\frac{b}{(a-b)}}{(s+a)} - \frac{\frac{a}{(a-b)}}{(s+b)} \quad (3.4)$$

$$C(t) = 1 + \frac{b}{(a-b)} e^{-at} - \frac{a}{(a-b)} e^{-bt} \quad (3.5)$$

Persamaan (3.5) menunjukkan bahwa jika $a \ll b$, maka respon dari sistem lup tertutup tersebut didominasi oleh e^{-at} dan efek dari e^{-bt} dapat diabaikan.

Dari karakteristik respon transien sistem lup tertutup yang diinginkan, dapat dihitung nilai pole dominan yang sesuai.

$$t_s(\pm 2\%) = 4\tau^* = 1 \text{ detik}$$

$$\tau^* = 0.25$$

$$a_i = \frac{1}{\tau^*} = 4$$

$$s_{dom} = s_{1roll} = s_{1pitch} = s_{1yaw} = -4$$

Nilai s_2 untuk *roll*, *pitch* dan *yaw* masing-masing adalah -20, -700 and -80. Nilai ini ditentukan melalui percobaan dengan MATLAB dengan nilai minimum $5s_1$.

Dengan memasukkan masing-masing nilai s_1 dan s_2 pada Persamaan (3.3), maka diperoleh model referensi untuk gerak *roll*, *pitch* dan *yaw* sebagai berikut:

Model referensi untuk gerak *roll*:

$$\frac{x_7(s)}{U_2^*(s)} = G_1(s) = \frac{80}{s^2 + 24s + 80}$$

atau

$$\begin{bmatrix} \dot{x}_7(t) \\ \dot{x}_8(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -80 & -24 \end{bmatrix} \begin{bmatrix} x_7(t) \\ x_8(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 80 \end{bmatrix} U_2^*(t) \quad (3.6)$$

Model referensi untuk gerak *pitch*:

$$\frac{x_9(s)}{U_3^*(s)} = G_1(s) = \frac{2800}{s^2 + 704s + 2800}$$

atau

$$\begin{bmatrix} \dot{x}_9(t) \\ \dot{x}_{10}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -2800 & -704 \end{bmatrix} \begin{bmatrix} x_9(t) \\ x_{10}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 2800 \end{bmatrix} U_3^*(t) \quad (3.7)$$

Model referensi untuk gerak *yaw*:

$$\frac{x_{11}(s)}{U_4^*(s)} = G_3(s) = \frac{320}{s^2 + 84s + 320}$$

atau

$$\begin{bmatrix} \dot{x}_{11}(t) \\ \dot{x}_{12}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -320 & -74 \end{bmatrix} \begin{bmatrix} x_{11}(t) \\ x_{12}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 320 \end{bmatrix} U_4^*(t) \quad (3.8)$$

3.1.3 Input-State Feedback Linearization

Persamaan *input-state feedback linearization* yang dicari adalah persamaan yang mampu melinierisasi dan mengeliminasi intraksi dalam sistem gerak rotasi

serta membuat sistem lup tertutup hasil rancangan memiliki respon transien dengan karakteristik yang telah ditentukan. Untuk mendapatkan persamaan ini, langkah pertama adalah dengan mendapatkan persamaan *state* yang berhubungan dengan U_2^* , U_3^* , dan U_4^* dari model referensi sistem gerak rotasi.

Dari Persamaan (3.6) – (3.8) diperoleh persamaan \dot{x}_8 , \dot{x}_{10} , dan \dot{x}_{12} sebagai berikut:

$$\begin{aligned}\dot{x}_8(t) &= -80x_7(t) - 24x_8(t) + 80U_2^*(t) \\ \dot{x}_{10}(t) &= -2800x_9(t) - 704x_{10}(t) + 2800U_3^*(t) \\ \dot{x}_{12}(t) &= -320x_{11}(t) - 84x_{12}(t) + 320U_4^*(t)\end{aligned}\quad (3.9)$$

Dengan mensubstitusi Persamaan (3.9) ke Persamaan (3.2), maka diperoleh persamaan *input-state feedback linearization* U_2 , U_3 , dan U_4 untuk gerak *roll*, *pitch* dan *yaw* sebagai berikut:

$$U_2(t) = \frac{J_{xx}}{l} \left(\frac{J_{zz} - J_{yy}}{J_{xx}} x_{10}(t) x_{12}(t) - 80x_7(t) - 24x_8(t) + 80U_2^*(t) \right) \quad (3.10)$$

$$U_3(t) = \frac{J_{yy}}{l} \left(\frac{J_{xx} - J_{zz}}{J_{yy}} x_8(t) x_{12}(t) - 2800x_9(t) - 704x_{10}(t) + 2800U_3^*(t) \right) \quad (3.11)$$

$$U_4(t) = J_{zz} \left(\frac{J_{yy} - J_{xx}}{J_{zz}} x_8(t) x_{10}(t) - 320x_{11}(t) - 84x_{12}(t) + 320U_4^*(t) \right) \quad (3.12)$$

Sistem lup tertutup gerak rotasi sebagaimana Persamaan 3.13 dapat diperoleh dengan mensubstitusi Persamaan 3.10 – 3.12 ke dalam persamaan 3.2.

$$\begin{aligned}\dot{x}_7 &= x_8 \\ \dot{x}_8 &= -80x_7 - 24x_8 + 80U_2^* \\ \dot{x}_9 &= x_{10} \\ \dot{x}_{10} &= -2800x_9 - 704x_{10} + 2800U_3^* \\ \dot{x}_{11} &= x_{12} \\ \dot{x}_{12} &= -320x_{11} - 84x_{12} + 320U_4^*\end{aligned}\quad (3.13)$$

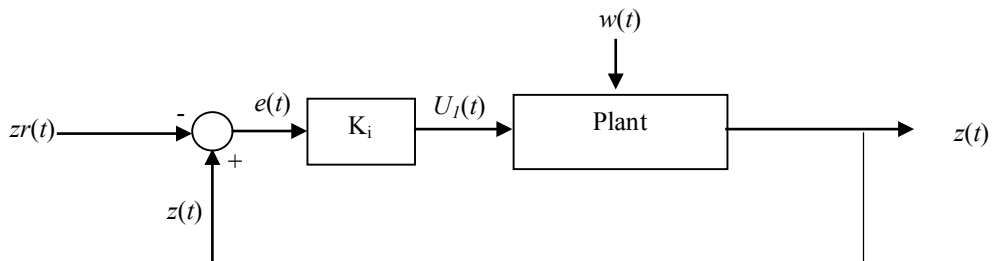
3.2 Sistem Gerak Translasi

Sistem gerak translasi dibagi menjadi 2 subsistem, yaitu subsistem *altitude* dan subsistem posisi. Subsistem *altitude* dinyatakan sebagai gerak translasi di sumbu z sedangkan subsistem posisi dinyatakan sebagai gerak translasi di sumbu x dan y. Masing-masing subsistem dikontrol dengan menggunakan metode kontrol

yang berbeda. Penelitian ini hanya difokuskan pada subsistem *altitude* karena merupakan aspek terpenting dalam mekanisme VTOL. Kontroler untuk subsistem *altitude* dirancang menggunakan metode fuzzy T-S sedangkan untuk subsistem posisi menggunakan kontroler Proporsional Derivatif (PD).

3.2.1 Perancangan Kontroler untuk *Altitude*

Untuk gerak translasi di sumbu Z, kontroler dirancang menggunakan metode fuzzy T-S. Sistem akan direpresentasikan kembali dengan pemodelan fuzzy T-S dengan titik operasi sebagai variabel premis dan model linier sebagai konsekuen. Dengan variabel premis yang sama, dirancang kontroler fuzzy T-S dengan sinyal kontrol sebagai konsekuennya. Skema PDC diterapkan sehingga setiap model linier akan dikompensasi oleh nilai penguat pada kontroler. Model referensi dirancang untuk membantu sistem mengikuti sinyal referensi yang diberikan. Kemudian dibentuk sistem augmented dari hasil pemodelan, kontroler dan model referensi. Nilai penguat kontroler dihitung dengan mempertimbangkan kriteria kestabilan Lyapunov dan performa H_∞ . Diagram blok sistem kontrol yang diterapkan pada sistem gerak translasi sebagaimana ditunjukkan di Gambar 3.3.



Gambar 3.3 Diagram blok sistem kontrol dengan fuzzy T-S.

3.2.1.1 Model Matematika *Altitude*

Dari Persamaan (2.23), diperoleh model matematika subsistem *altitude* quadcopter sebagai berikut:

$$\ddot{z} = -g + (\cos\phi \cos\theta) \frac{U_1}{m} \quad (3.14)$$

Atau dalam bentuk *state-space*:

$$\begin{bmatrix} \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_5 \\ x_6 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m}(\cos x_7 \cos x_9) \end{bmatrix} U_1 + G$$

dengan

$$x_5 = z; \quad x_6 = \dot{z}; \quad G = \begin{bmatrix} 0 & -g \end{bmatrix}^T$$

3.2.1.2 Pemodelan Sistem Gerak Translasi dengan Fuzzy T-S

Model matematika sistem gerak translasi di sumbu z sebagaimana ditunjukkan Persamaan (3.14) direpresentasikan kembali dalam bentuk model fuzzy T-S. Langkah pertama yang dilakukan adalah linierisasi sistem. Sistem dilinierisasi di titik operasi ketika quadcopter melakukan mekanisme VTOL. Titik operasinya adalah sekitar 0 rad untuk sudut *roll*, *pitch* dan *yaw*. Pada penelitian ini, batasan titik operasi yang digunakan adalah 0 rad dan $\pm 0,174$ rad. Proses linierisasi dilakukan dengan mendapatkan matriks Jacobian dari sistem di titik operasi yang ditentukan sebagai berikut:

$$\text{Titik operasi I: } z_I^* = \begin{bmatrix} 0 & 0 \end{bmatrix}^T$$

$$\text{Model linier I: } \dot{x} = A_1 x + B_1 U_1 + G$$

dengan

$$A_1 = A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}; \quad B_1 = \begin{bmatrix} 0 \\ \frac{1}{3.5} \end{bmatrix}$$

$$\text{Titik operasi II: } z_2^* = \begin{bmatrix} 0 & \pm 0,174 \end{bmatrix}^T$$

$$\text{Model linier II: } \dot{x} = A_2 x + B_2 U_1 + G$$

dengan

$$A_2 = A; \quad B_2 = \begin{bmatrix} 0 \\ \frac{0.985}{3.5} \end{bmatrix}$$

$$\text{Titik operasi III: } z_3^* = \begin{bmatrix} \pm 0,174 & 0 \end{bmatrix}^T$$

$$\text{Model linier III: } \dot{x} = A_3 x + B_3 U_1 + G$$

dengan

$$A_3 = A \quad ; \quad B_3 = \begin{bmatrix} 0 \\ \frac{0.985}{3.5} \end{bmatrix}$$

Titik operasi IV: $z_4^* = [\pm 0,174 \pm 0,174]^T$

Model linier III: $\dot{x} = A_4 x + B_4 U_1 + G$

dengan

$$A_4 = A \quad ; \quad B_4 = \begin{bmatrix} 0 \\ \frac{0.970}{3.5} \end{bmatrix}$$

Matriks keluaran

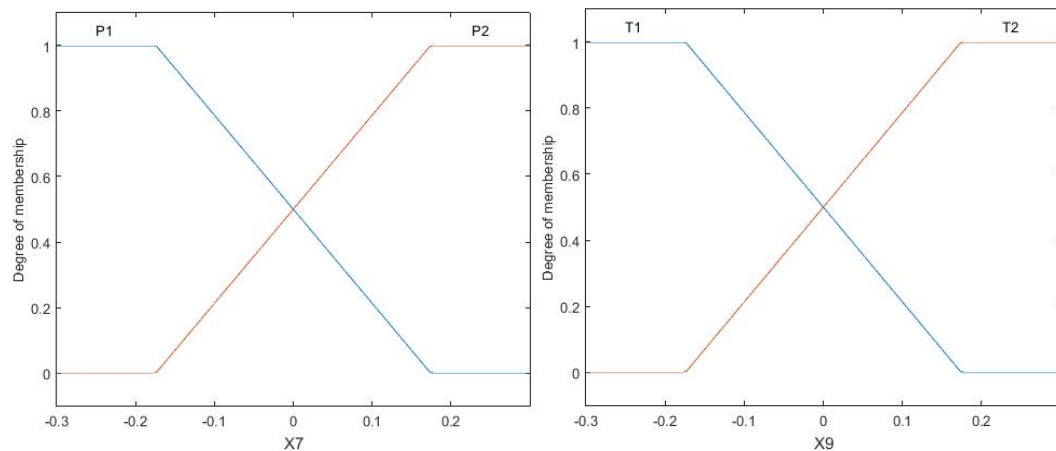
$$C_1 = C_2 = C_3 = C_4 = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

$$D_1 = D_2 = D_3 = D_4 = 0$$

Model linier yang didapatkan dirangkai menjadi aturan *fuzzy*. Titik kerja yang digunakan untuk linierisasi merupakan variabel premis, sedangkan konsekuennya adalah model linier yang didapatkan.

$$\begin{array}{ll} \text{If} & x_7(t) \text{ is } P_i \text{ and } x_9 \text{ is } T_i \\ \text{Then} & \dot{x}(t) = A_j x(t) + B_j u(t) + w(t) \\ & y(t) = C_j x(t); \quad i = 1,2; \quad j = 1,2,3,4 \end{array} \quad (3.15)$$

P_i dan T_i adalah fungsi keanggotaan berupa fungsi trapesium sebagaimana ditunjukkan pada Gambar 3.4 dan $w(t)$ adalah gangguan.



Gambar 3.4 Fungsi keanggotaan

Mekanisme inferensi fuzzy T-S menggunakan penghubung *AND* sedangkan proses defuzzifikasi menggunakan metode *center average*. Secara keseluruhan, model plant fuzzy dapat dituliskan dalam Persamaan (3.16).

Model plant secara keseluruhan

$$\begin{aligned}\dot{x}(t) &= \sum_1^j h_j [A_j x(t) + B_j u(t)] + w(t) \\ y(t) &= C_j x(t); \quad j = 1, 2, 3, 4\end{aligned}\tag{3.16}$$

dengan

$$\begin{aligned}h_j &= \frac{\mu_j(x(t))}{\sum_{j=1}^4 \mu_j(x(t))} \\ \mu_{1,2}(x(t)) &= \min(P_1(x_7(t)), T_{1,2}(x_9(t))) \\ \mu_{3,4}(x(t)) &= \min(P_2(x_7(t)), T_{1,2}(x_9(t))) \\ x(t) &= [x_7(t), x_9(t)]\end{aligned}$$

3.2.1.3 Kontroler Fuzzy T-S

Aturan kontroler fuzzy T-S dirancang dengan variabel premis sama dengan variabel premis pada aturan model fuzzy T-S dan sebagai konsekuennya adalah sinyal kontrol. Skema PDC diterapkan pada kontroler sehingga setiap model linier pada model fuzzy T-S akan dikompensasi dengan nilai penguat yang sesuai.

Aturan kontroler fuzzy T-S:

$$\begin{aligned}\text{if} \quad & x_7(t) \text{ is } P_i \text{ and } x_9(t) \text{ is } T_i \\ \text{then} \quad & u(t) = K_j * e(t)\end{aligned}\tag{3.17}$$

dengan

$$\begin{aligned}e(t) &= x(t) - x_r(t) \\ i &= 1, 2 \\ j &= 1, 2, 3, 4\end{aligned}$$

$x_r(t)$ adalah *state* referensi dan $x(t)$ adalah *state* aktual.

3.2.1.4 Model Referensi untuk Sistem Gerak Translasi

Untuk membantu sistem gerak translasi mampu mengikuti sinyal referensi yang diberikan, dirancang model referensi dengan metode sama dengan perancangan model referensi pada sistem gerak rotasi.

$$\frac{x_s(s)}{U_1^*} = \frac{8}{s^2 + 10.8s + 8} \quad (3.18)$$

atau dalam bentuk persamaan *state-space*:

$$\dot{x}(t) = A_r x(t) + B_r u(t)$$

dengan

$$A_r = \begin{bmatrix} 0 & 1 \\ -8 & -10.8 \end{bmatrix}; \quad B_r = \begin{bmatrix} 0 \\ 8 \end{bmatrix}$$

3.2.1.5 Perancangan Sistem Augmented

Dari model, kontroler fuzzy T-S dan model referensi yang telah dirancang, dibuat sistem *augmented* sebagai berikut:

$$\dot{x}(t) = \sum_1^j h_j [\bar{A}_j \bar{x}(t) + \bar{E}_j \bar{w}(t)] \quad (3.19)$$

dengan

$$\bar{A}_j = \begin{bmatrix} A + B_j K_j & -B_j K_j \\ 0 & A_r \end{bmatrix}; \quad \bar{x}(t) = \begin{bmatrix} x(t) \\ x_r(t) \end{bmatrix}$$

$$\bar{E}_j = \begin{bmatrix} I & 0 \\ 0 & B_r \end{bmatrix}; \quad \bar{x}(t) = \begin{bmatrix} w(t) \\ r(t) \end{bmatrix}$$

3.2.1.6 Kriteria Kestabilan Lyapunov

Untuk menjamin kestabilan sistem lup tertutup yang dirancang, digunakan kriteria kestabilan Lyapunov.

Sistem lup tertutup:

$$\dot{x}(t) = \sum_1^j h_j \bar{A}_j \bar{x}(t) \quad (3.20)$$

Normalisasi diterapkan derajat keanggotaan fuzzy T-S sehingga:

$$\sum_1^j h_j = 1$$

Turunan pertama fungsi Lyapunov:

$$\begin{aligned}\dot{V}(x) &= (\bar{A}_j \bar{x}(t))^T \bar{P} \bar{x}(t) + \bar{x}(t)^T \bar{P} (\bar{A}_j \bar{x}(t)) \\ &= \bar{x}(t)^T (\bar{A}_j^T \bar{P} + \bar{P} \bar{A}_j) \bar{x}(t)\end{aligned}\quad (3.21)$$

dengan

$$\bar{P} = \bar{P}^T = \begin{bmatrix} P_{11} & 0 \\ 0 & P_{22} \end{bmatrix} > 0$$

Dari Persamaan (3.21), dapat disimpulkan bahwa kestabilan sistem lup tertutup pada Persamaan (3.20) dapat terjamin jika terdapat \bar{P} yang memenuhi syarat berikut:

1. $\bar{P} > 0$
2. $\bar{A}_j^T \bar{P} + \bar{P} \bar{A}_j < 0$

3.2.1.7 Performa H_∞

Dalam penelitian ini, kontroler dirancang juga untuk mengatasi gangguan terbatas yang diberikan pada sistem gerak translasi. Untuk itu, digunakan performa H_∞ sehingga segala efek yang ditimbulkan oleh gangguan dapat dilemahkan sampai sama atau kurang dari level pelemahan yang ditentukan.

$$\int_0^{t_f} [x(t) - x_r(t)]^T Q [x(t) - x_r(t)] dt \leq \rho^2 \int_0^{t_f} w(t)^T w(t) dt \quad (3.22)$$

Untuk sistem *augmented* pada Persamaan (3.19), Pertidaksamaan (3.22) dapat dituliskan sebagai:

$$\int_0^{t_f} \bar{x}(t)^T \bar{Q} \bar{x}(t) dt \leq \rho^2 \int_0^{t_f} w(t)^T w(t) dt \quad (3.23)$$

dengan

$$\bar{Q} = \begin{bmatrix} Q & -Q \\ -Q & Q \end{bmatrix}$$

Pertidaksamaan (3.23) dapat dimodifikasi menjadi Pertidaksamaan (3.24) dengan memasukkan sistem augmented pada Persamaan (3.19). [18]

$$\int_0^{t_f} \bar{x}(t)^T \bar{Q} \bar{x}(t) dt \leq \bar{x}(0)^T \bar{P} \bar{x}(0) + \int_0^{t_f} \left\{ \sum_{j=1}^j h_j \bar{x}(t)^T [\bar{Q} + \bar{A}_j^T \bar{P}_j + \bar{P} \bar{A}_j + \frac{1}{\rho^2} \bar{P} \bar{E}_j \bar{E}_j^T \bar{P}] \bar{x}(t) + \rho^2 \bar{w}(t)^T \bar{w}(t) \right\} dt \quad (3.24)$$

Dari Pertidaksamaan (3.24) dapat disimpulkan bahwa sistem *augmented* akan memenuhi performa H^∞ jika pertidaksamaan (3.24) dapat dipenuhi.

$$\bar{Q} + \bar{A}_j^T \bar{P} + \bar{P} \bar{A}_j + \frac{1}{\rho^2} \bar{P} \bar{E}_j \bar{E}_j^T \bar{P} < 0 \quad (3.24)$$

3.2.1.8 Penghitungan Matriks P dan Nilai Penguat K_i

Untuk mendapatkan matriks \bar{P} dan K_j yang memenuhi Pertidaksamaan (3.24), dilakukan langkah-langkah matematis sebagai berikut:

1. Substitusi \bar{P} , \bar{Q} , \bar{A}_j , and \bar{E}_j dari Persamaan (3.19) ke Persamaan (3.24) dan transformasikan ke dalam bentuk LMI sehingga diperoleh:

$$\begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix} < 0 \quad (3.25)$$

dengan

$$S_{11} = Q + (A_j + B_j K_j)^T P_{11} + P_{11} (A_j + B_j K_j) + \frac{1}{\rho^2} P_{11} P_{11}$$

$$S_{12} = -Q + P_{11} (-B_j K_j)$$

$$S_{21} = -Q + (-B_j K_j)^T P_{11}$$

$$S_{22} = Q + A_r^T P_{22} + P_{22} A_r + \frac{1}{\rho^2} P_{22} B_r B_r^T P_{22}$$

2. Dengan metode *schur complement*, S_{11} dapat ditransformasikan ke dalam bentuk LMI sebagai berikut:

$$\begin{bmatrix} M_{11} & M_{12} & 0 \\ M_{21} & M_{22} & P_{22} B_r \\ 0 & B_r^T P_{22} & -\rho^2 I \end{bmatrix} < 0 \quad (3.26)$$

dengan

$$M_{11} = (A_j + B_j K_j)^T P_{11} + P_{11} (A_j + B_j K_j) + \frac{1}{\rho^2} P_{11} P_{11} + Q$$

$$M_{12} = M_{21}^T = -P_{11} B_j K_j - Q$$

$$M_{22} = A_r^T P_{22} + P_{22} A_r + Q$$

3. Untuk menyelesaikan Pertidaksamaan (3.26), pertama diselesaikan $M_{11} < 0$.

$$(A_j + B_j K_j)^T P_{11} + P_{11} (A_j + B_j K_j) + \frac{1}{\rho^2} P_{11} P_{11} + Q < 0 \quad (3.27)$$

dengan modifikasi sebagaimana dilakukan pada [23], Pertidaksamaan (3.27) dapat dimodifikasi menjadi:

$$W_{11} A_j^T + (B_j Y_j)^T + A_j W_{11} + B_j Y_j + \frac{1}{\rho^2} I + W_{11} Q W_{11} < 0 \quad (3.28)$$

dengan

$$W_{11} = P_{11}^{-1}$$

$$Y_j = K_j W_{11}$$

Pertidaksamaan (3.28) dapat ditransformasikan ke dalam bentuk LMI sebagai berikut:

$$\begin{bmatrix} H_{11} & W_{11} \\ W_{11} & -Q^{-1} \end{bmatrix} < 0 \quad (3.29)$$

dengan

$$H_{11} = W_{11} A_j^T + (B_j Y_j)^T + A_j W_{11} + B_j Y_j + \frac{1}{\rho^2} I$$

4. Dengan menentukan nilai matriks \bar{Q} dan level pelemahan (ρ), Pertidaksamaan (3.29) dapat diselesaikan dengan bantuan MATLAB sehingga diperoleh nilai matriks P_{11} dan K_j . Selanjutnya, nilai matriks P_{11} dapat diperoleh dengan menyelesaikan Pertidaksamaan (3.26).

$$\rho = 0.003$$

$$Q = \begin{bmatrix} 44 \times 10^{-10} & 0 \\ 0 & 37 \times 10^{-10} \end{bmatrix}$$

$$P_{11} = 10^{-7} \begin{bmatrix} 0.4247 & 0.3183 \\ 0.3183 & 0.3412 \end{bmatrix}$$

$$P_{22} = 10^{-5} \begin{bmatrix} 0.4489 & 0.0856 \\ 0.0856 & 0.2125 \end{bmatrix}$$

$$K_1 = \begin{bmatrix} -16.6100 & -16.6535 \end{bmatrix}$$

$$K_2 = \begin{bmatrix} -16.6981 & -16.7374 \end{bmatrix}$$

$$K_3 = \begin{bmatrix} -16.6981 & -16.7374 \end{bmatrix}$$

$$K_4 = \begin{bmatrix} -16.7877 & -16.8226 \end{bmatrix}$$

3.2.2 Perancangan Kontroler untuk Posisi

Sebagaimana disebutkan sebelumnya, bahwa keluaran dari output kontroler posisi adalah masukan untuk sistem gerak rotasi. Untuk itu, diperlukan transformasi dari U_x dan U_y menjadi θ_d dan ϕ_d . Kontroler yang dirancang untuk subsistem posisi x dan y adalah kontroler PD di mana parameter kontroler dihitung berdasarkan letak pole sistem yang ditentukan.

3.2.2.1 Transformasi Output U_x dan U_y Menjadi Input θ_d dan ϕ_d

Sebelum melakukan perancangan kontroler untuk gerak translasi di sumbu x dan sumbu y , sebagaimana ditunjukkan Gambar x.x bahwa keluaran dari kontroler sistem gerak translasi di sumbu x dan y yaitu U_x dan U_y akan menjadi input referensi bagi gerak *pitch* (θ_d) dan *roll* (ϕ_d). Untuk itu perlu dilakukan transformasi dari U_x dan U_y menjadi θ_d dan ϕ_d . Berikut adalah langkah-langkah transformasi yang dilakukan. [28]

Dari Persamaan 2.15 diperoleh dinamika gerak translasi quadcopter sebagai berikut:

$$\begin{aligned} \sum F &= m\ddot{\Gamma} \\ R_{\Theta}U_T + F_g z_g &= m\ddot{\Gamma} \end{aligned} \quad (3.30)$$

Ditentukan matriks $U = [U_x \ U_y \ U_z]$ dengan:

$$U = \ddot{\Gamma}$$

sehingga diperoleh persamaan berikut:

$$\begin{aligned} R_{\Theta}U_T + F_g z_g &= mU \\ U_T &= R_{\Theta}^T (mU - F_g z_g) \end{aligned} \quad (3.31)$$

Dengan mensubstitusi Persamaan 2.20 ke dalam Persamaan 3.31 diperoleh:

$$\frac{1}{m}U_T = R_{\Theta}^T (U + g z_g) \quad (3.31)$$

Substitusi Persamaan 2.17 dan 2.19 ke dalam Persamaan 3.31:

$$\frac{1}{m} \begin{bmatrix} 0 \\ 0 \\ U_1 \end{bmatrix} = \begin{bmatrix} C\theta C\psi & C\theta S\psi & -S\theta \\ S\phi S\theta C\psi - C\phi S\psi & S\phi S\theta S\psi + C\phi C\psi & S\phi C\theta \\ C\phi S\theta C\psi + S\phi S\psi & C\phi S\theta S\psi - S\phi C\psi & C\phi C\theta \end{bmatrix} \begin{bmatrix} U_x \\ U_y \\ U_z + g \end{bmatrix} \quad (3.32)$$

Dengan operasi sederhana pada Persamaan 3.32, diperoleh persamaan sebagai berikut:

$$U_x C\theta C\psi + U_y C\theta S\psi - (U_z + g)S\theta = 0 \quad (3.33)$$

$$U_x (S\phi S\theta C\psi - C\phi S\psi) + U_y (S\phi S\theta S\psi + C\phi C\psi) - (U_z + g)S\phi C\theta = 0 \quad (3.34)$$

$$U_x (C\phi S\theta C\psi + S\phi S\psi) + U_y (C\phi S\theta S\psi - S\phi C\psi) - (U_z + g)C\phi C\theta = \frac{U_1}{m} \quad (3.35)$$

Persamaan θ_d dapat diperoleh dengan membagi Persamaan 3.33 dengan $\cos \theta$ sehingga diperoleh persamaan berikut:

$$U_x \cos\psi + U_y \sin\psi - (U_z + g)\tan\theta = 0$$

$$\tan\theta = \frac{U_x \cos\psi + U_y \sin\psi}{(U_z + g)}$$

$$\theta_d = \arctan\left(\frac{U_x \cos\psi + U_y \sin\psi}{(U_z + g)}\right) \quad (3.36)$$

Persamaan ϕ dapat diperoleh dengan langkah-langkah sebagai berikut:

1. Persamaan 3.34 dikalikan dengan $\cos \phi$ sehingga diperoleh:

$$U_x (\sin\phi \sin\theta \cos\psi - \cos\phi \sin\psi) \cos\phi + U_y (\sin\phi \sin\theta \sin\psi + \cos\phi \cos\psi) \cos\phi - (U_z + g) \sin\phi \cos\theta \cos\phi = 0 \quad (3.37)$$

2. Persamaan 3.35 dikalikan dengan $\sin \phi$ sehingga diperoleh:

$$U_x (\cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi) \sin\phi + U_y (\cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi) \sin\phi - (U_z + g) \cos\phi \cos\theta \sin\phi = \frac{U_1}{m} \sin\phi \quad (3.38)$$

3. Persamaan 3.38 dikurangi Persamaan 3.37, diperoleh:

$$U_x (\sin^2\phi \sin\psi + \cos^2\phi \sin\psi) + U_y (-\sin^2\phi \cos\psi - \cos^2\phi \cos\psi) = \frac{U_1}{m} \sin\phi$$

$$U_x \sin\psi + U_y \cos\psi = \frac{U_1}{m} \sin\phi \quad (3.39)$$

4. Dari Persamaan 3.31 diketahui bahwa:

$$\left(\frac{1}{m} U_T\right)^T \left(\frac{1}{m} U_T\right) = (U + g z_g)^T (U + g z_g)$$

$$\begin{aligned}\left(\frac{U_1}{m}\right)^2 &= U_x^2 + U_y^2 + U_z^2 \\ \frac{U_1}{m} &= \sqrt{U_x^2 + U_y^2 + U_z^2}\end{aligned}\quad (3.40)$$

5. Dengan mensubstitusi Persamaan 3.40 ke dalam Persamaan 3.39, diperoleh persamaan ϕ_d sebagai berikut:

$$U_x \sin \psi + U_y \cos \psi = \sqrt{U_x^2 + U_y^2 + U_z^2} \sin \phi$$

$$\sin \phi = \frac{U_x \sin \psi + U_y \cos \psi}{\sqrt{U_x^2 + U_y^2 + U_z^2}}$$

$$\phi_d = \arcsin \left(\frac{U_x \sin \psi + U_y \cos \psi}{\sqrt{U_x^2 + U_y^2 + U_z^2}} \right)$$

3.2.2.2 Perancangan Kontroler untuk Gerak Translasi di Sumbu X

Dari Persamaan (2.21), diperoleh model matematika subsistem gerak translasi di sumbu x sebagai berikut:

$$\ddot{x} = (\sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi) \frac{U_1}{m} \quad (3.41)$$

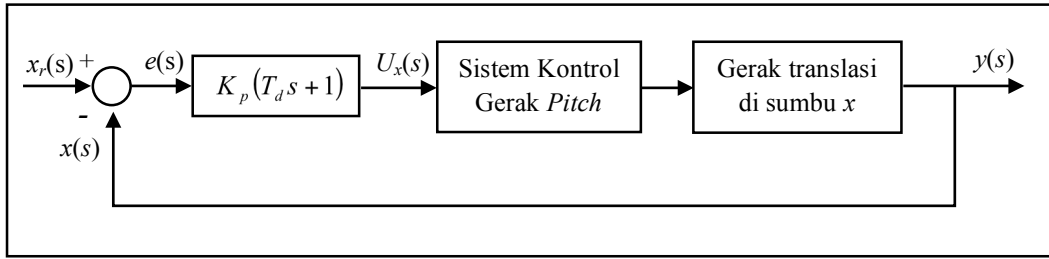
atau dalam bentuk persamaan *state-space*:

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= (\sin x_7 \sin x_{11} + \cos x_7 \sin x_9 \cos x_{11}) \frac{U_1}{m}\end{aligned}\quad (3.42)$$

Kontroler dirancang sehingga quadcopter dapat mempertahankan dan/atau mencapai posisi yang ditentukan di sumbu x . Sinyal kontrol U_x yang dihasilkan ditransformasikan sebagai input θ yang dibahas di Subbab 3.2.2.1. Dengan adanya transformasi tersebut, subsistem gerak translasi di sumbu x dapat dinyatakan sebagai berikut:

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= U_x\end{aligned}\quad (3.43)$$

Sistem kontrol gerak translasi di sumbu x dirancang menggunakan metode PD dengan diagram blok sebagaimana ditunjukkan Gambar 3.5.



Gambar 3.5 Diagram blok sistem kontrol gerak translasi di sumbu x .

Dari Gambar 3.5 dapat diperoleh persamaan sistem lup tertutup sebagai berikut:

$$\frac{y(s)}{x_r(s)} = \frac{\frac{K_p(T_d s + 1)}{s^2}}{1 + \frac{K_p(T_d s + 1)}{s^2}} = \frac{K_p(T_d s + 1)}{s^2 + K_p(T_d s + 1)} = \frac{T_d s + 1}{\frac{1}{K_p} s^2 + T_d s + 1}$$

Nilai parameter kontroler K_p dan T_d didasarkan pada karakteristik respon transien yang diinginkan. Karakteristik yang diinginkan sama dengan karakteristik pada sistem gerak rotasi. Namun, karena sistem kontrol posisi merupakan *outter loop* maka t_s harus lebih besar dari t_s sistem kontrol gerak rotasi. Karakteristik ini dapat diperoleh dengan penetapan nilai pole sebagai berikut:

$$t_s (\pm 2\%) = 4\tau^* = 7 \text{ detik}$$

$$\tau^* = 1.75$$

$$s_{dom} = s_{x1} = -\frac{1}{\tau^*} = -0.6$$

Nilai s_{x2} adalah -10. Nilai ini ditentukan melalui percobaan dengan MATLAB dengan nilai minimum $5s_1$. Dari pasangan nilai pole ini dapat diperoleh nilai parameter kontroler K_p dan T_d .

$$\text{pole I: } s_{x1} = -0.6 \quad \text{pole II: } s_{x2} = -168$$

Persamaan karakteristik sistem yang diinginkan:

$$\begin{aligned} & s^2 + (\text{pole I} + \text{pole II})s + (\text{pole I} \times \text{pole II}) \\ &= s^2 + 168,6s + 100,8 \\ &= \frac{1}{100,8}s^2 + \frac{168,6}{100,8}s + 1 \end{aligned}$$

Persamaan karakteristik sistem:

$$\frac{1}{K_p} s^2 + T_d s + 1$$

Perhitungan nilai K_p dan T_d :

$$\frac{1}{K_p} s^2 + T_d s + 1 = \frac{1}{100,8} s^2 + \frac{168,6}{100,8} s + 1$$

$$K_p = 100,8$$

$$T_d = 1,67$$

3.2.2.3 Perancangan Kontroler untuk Gerak Translasi di Sumbu Y

Dari Persamaan (2.22), diperoleh model matematika subsistem gerak translasi di sumbu y sebagai berikut:

$$\ddot{y} = (-\sin \phi \sin \psi + \cos \phi \sin \theta \sin \psi) \frac{U_1}{m} \quad (3.44)$$

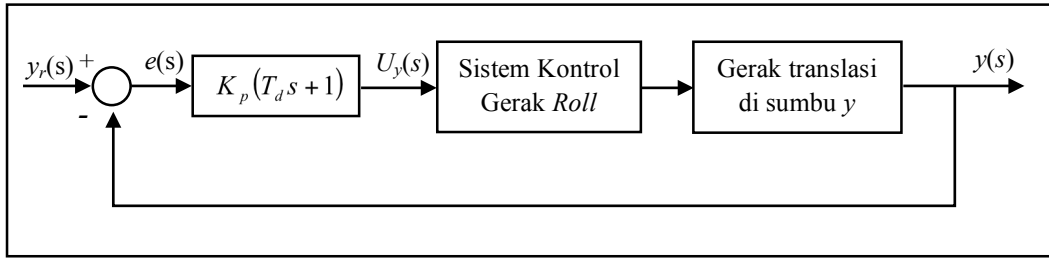
atau dalam bentuk persamaan state-space:

$$\begin{aligned} \dot{x}_3 &= x_4 \\ \dot{x}_4 &= (-\sin x_7 \sin x_{11} + \cos x_7 \sin x_9 \sin x_{11}) \frac{U_1}{m} \end{aligned} \quad (3.45)$$

Kontroler dirancang sehingga quadcopter mempertahankan dan/atau mencapai posisi yang ditentukan di sumbu y. Sinyal kontrol U_y yang dihasilkan ditransformasikan sebagai input ϕ yang dibahas di subbab sebelumnya. Dengan adanya transformasi tersebut, subsistem gerak translasi di sumbu x dapat dinyatakan sebagai berikut:

$$\begin{aligned} \dot{x}_3 &= x_4 \\ \dot{x}_4 &= U_y \end{aligned} \quad (3.46)$$

Sistem kontrol gerak translasi di sumbu y dirancang menggunakan metode PD dengan diagram blok sebagaimana ditunjukkan Gambar 3.6.



Gambar 3.6 Diagram blok sistem kontrol gerak translasi di sumbu y.

Dari Gambar 3.6 dapat diperoleh persamaan sistem lup tertutup sebagai berikut:

$$\frac{y(s)}{y_r(s)} = \frac{\frac{K_p(T_d s + 1)}{s^2}}{1 + \frac{K_p(T_d s + 1)}{s^2}} = \frac{K_p(T_d s + 1)}{s^2 + K_p(T_d s + 1)} = \frac{T_d s + 1}{\frac{1}{K_p} s^2 + T_d s + 1}$$

Nilai parameter kontroler K_p dan T_d didasarkan pada karakteristik respon transien yang diinginkan. Karakteristik yang diinginkan sama dengan karakteristik pada sistem gerak rotasi. Namun, karena sistem kontrol posisi merupakan *outter loop* maka t_s harus lebih besar dari t_s sistem kontrol gerak rotasi. Karakteristik ini dapat diperoleh dengan penetapan nilai pole sebagai berikut:

$$t_s(\pm 2\%) = 4\tau^* = 7 \text{ detik}$$

$$\tau^* = 1.75$$

$$s_{dom} = s_{x1} = -\frac{1}{\tau^*} = -0.6$$

Nilai s_{x2} adalah -10. Nilai iniditentukan melalui percobaan dengan MATLAB dengan nilai minimum $5s_1$. Dari pasangan nilai pole ini dapat diperoleh nilai parameter kontroler K_p dan T_d .

$$\text{pole I: } s_{x1} = -0.6$$

$$\text{pole II: } s_{x2} = -168$$

Persamaan karakteristik sistem yang diinginkan:

$$s^2 + (\text{pole I} + \text{pole II})s + (\text{pole I} \times \text{pole II})$$

$$= s^2 + 168,6s + 100,8$$

$$= \frac{1}{100,8} s^2 + \frac{168,6}{100,8} s + 1$$

Persamaan karakteristik sistem:

$$\frac{1}{K_p} s^2 + T_d s + 1$$

Perhitungan nilai K_p dan T_d :

$$\frac{1}{K_p} s^2 + T_d s + 1 = \frac{1}{100,8} s^2 + \frac{168,6}{100,8} s + 1$$

$$K_p = 100,8$$

$$T_d = 1,67$$

Halaman ini sengaja dikosongkan

BAB 4

HASIL DAN PEMBAHASAN

Untuk menguji sistem kontrol hasil desain pada BAB 3, dilakukan simulasi dengan bantuan MATLAB. Masing-masing sistem kontrol pada gerak rotasi dan gerak translasi disimulasikan dengan dua kondisi, yaitu kondisi ideal tanpa gangguan dan kondisi dengan gangguan. Nilai error dalam simulasi pada kedua kondisi tersebut dihitung menggunakan metode *Integral Squared Error*(ISE).

4.1 Simulasi Sistem Kontrol dalam Kondisi Tanpa Gangguan

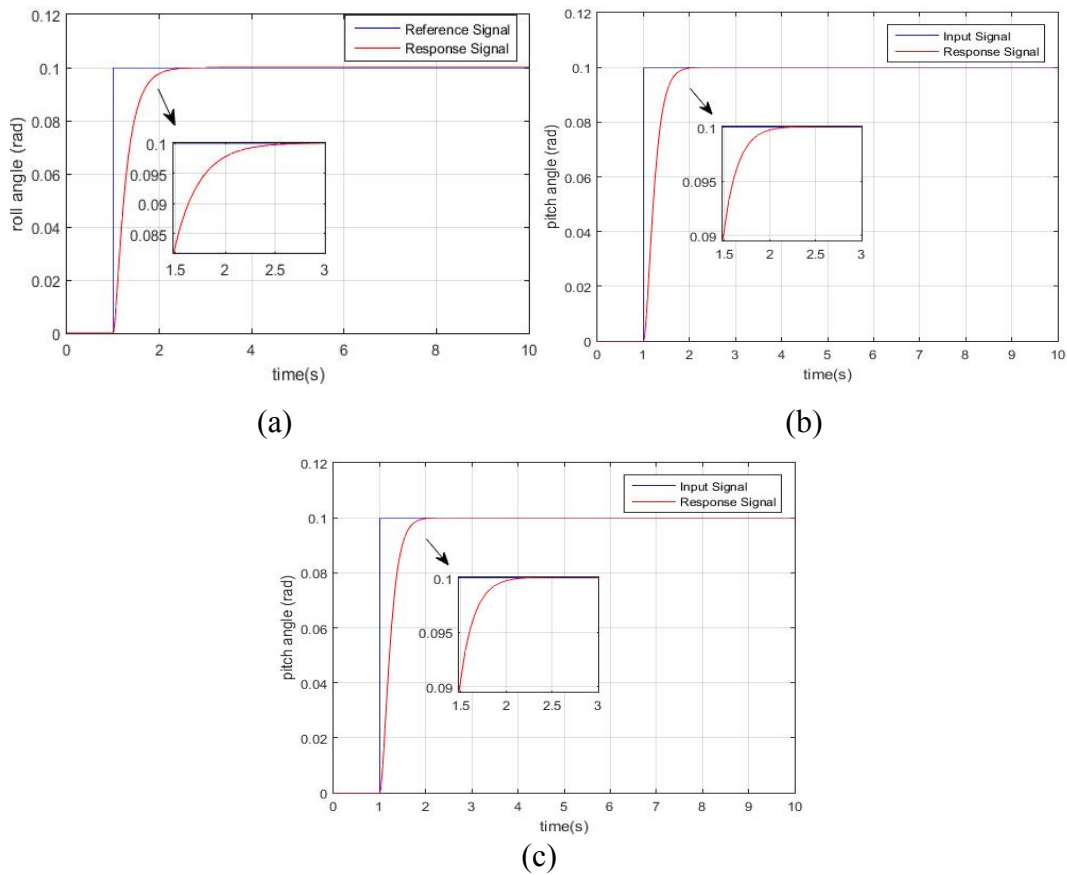
Sebelum simulasi sistem quadcopter dibuat, pertama dilakukan simulasi pada sistem gerak rotasi. Hal ini dilakukan untuk mengetahui apakah karakteristik respon transien sistem gerak rotasi sesuai dengan yang telah ditentukan saat perancangan atau tidak.

Simulasi dilakukan dengan memberikan sinyal masukan berupa sinyal unit step pada sistem kontrol gerak rotasi. Pemilihan sinyal masukan ini dilakukan untuk menyesuaikan dengan perancangan. Hasil simulasi sistem kontrol gerak rotasi dapat dilihat pada Gambar 4.1.

Sebagaimana disebutkan pada BAB 3, bahwa karakteristik respon transien sistem gerak rotasi yang diinginkan adalah:

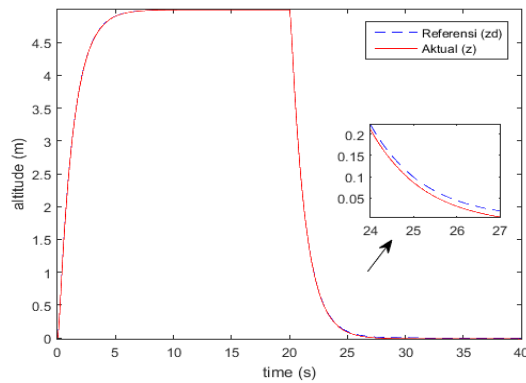
- a) *Zero overshoot*;
- b) *zero offset*; dan
- c) *settling time* (t_s) 1 detik $\pm 2\%$.

Dari Gambar 4.1a dapat dilihat bahwa respon sudut roll telah memenuhi karakteristik yang diinginkan dengan t_s sebesar 1 detik. Begitu juga dengan respon sudut yaw sebagaimana dapat dilihat pada Gambar 4.1c. Respon sudut yaw sesuai dengan karakteristik yang diinginkan. Namun sedikit berbeda dengan respon sudut pitch sebagaimana pada Gambar 4.1b. Respon sudut pitch memiliki t_s sebesar 0,8 detik. Hal ini tidak menjadi masalah karena target t_s respon sistem gerak rotasi adalah 1 detik. Dengan hasil tersebut dapat disimpulkan bahwa respon sistem gerak rotasi telah memenuhi karakteristik yang diinginkan



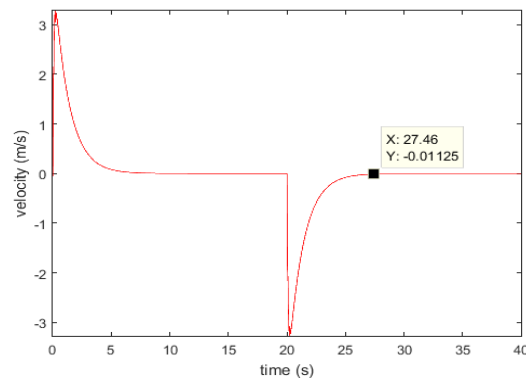
Gambar 4.1 Respon sistem gerak rotasi terhadap sinyal unit step: (a) gerak *roll*, (b) gerak *pitch* dan (c) gerak *yaw*.

Selanjutnya adalah simulasi sistem quadcopter secara menyeluruh pada kondisi tanpa gangguan. Pada sistem gerak translasi, sinyal masukan untuk subsistem *altitude* dan posisi dibuat berbeda. Sinyal referensi untuk sistem kontrol *altitude* adalah sinyal respon model referensi terhadap sinyal pulsa dengan *offset* 0 dan amplitudo 5. Sedangkan sinyal masukan untuk sistem kontrol posisi *x* dan *y* adalah 0. Hal ini dilakukan untuk mensimulasikan mekanisme VTOL di titik koordinat (0,0) dengan ketinggian maksimal 5m. Hasil simulasi sebagaimana pada Gambar 4.2 menunjukkan bahwa respon mampu mengikuti referensi yang diberikan dengan nilai ISE *altitude* sebesar 0.0037 m. Penyimpangan 0.013 m mulai terlihat pada detik 25.03 dan terjadi *offset* sebesar -0.014 m.



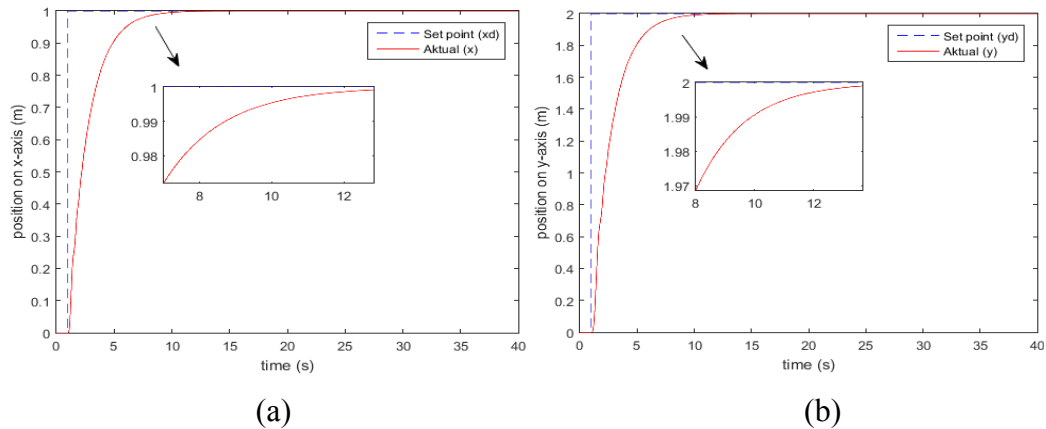
Gambar 4.2 Respon *tracking* pada *altitude*

Pada mekanisme VTOL, hal yang juga harus diperhatikan adalah kecepatan vertikal ketika quadcopter menyentuh titik 0 sumbu z . Hasil respon kecepatan vertikal dapat dilihat pada Gambar 4.3. Dari Gambar 4.2 diketahui bahwa respon altitude mencapai titik 0 pada detik ke 27,46. Jika dilihat pada Gambar 4.3, pada detik tersebut kecepatan vertikal cukup kecil dan aman yaitu -0.01125 m/s.



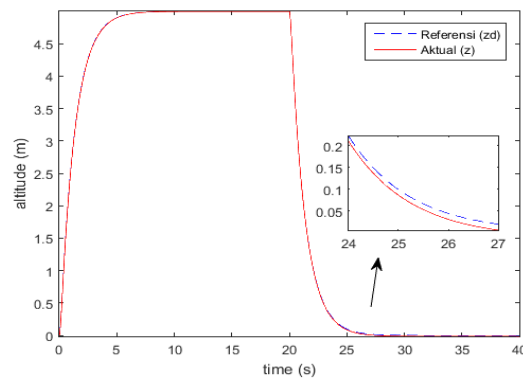
Gambar 4.3 Respon kecepatan vertikal

Selanjutnya adalah simulasi VTOL dengan penambahan perubahan posisix dan y dari titik (0,0) menuju (1,2). Respon posisi x dan y sebagaimana ditunjukkan pada Gambar 4.4. Pada Gambar 4.4a dapat dilihat bahwa respon posisi x dapat mencapai *set point* yang ditentukan yaitu 1 m dengan *settling time* (t_s) sebesar 7 detik. Hal ini berarti respon posisi x telah sesuai dengan kriteria perancangan. Begitu juga dengan respon posisi y . Gambar 4.4b menunjukkan bahwa respon posisi y mampu mencapai *set point* yang ditentukan yaitu 2 m dengan t_s sebesar 7 detik.



Gambar 4.4 Respon posisi (a) posisi x dan (b) posisi y

Tujuan ditambahkan perubahan posisi x dan y pada saat mekanisme VTOL adalah untuk melihat pengaruhnya pada respon sistem kontrol altitude yang dirancang. Sebagaimana disebutkan pada bab sebelumnya bahwa keluaran dari kontroler posisi x dan y akan ditransformasikan menjadi sinyal referensi ϕ_r dan θ_r untuk sistem gerak rotasi sehingga perubahan posisi x dan y akan mempengaruhi respon sudut *roll* dan *pitch* pada sistem gerak rotasi. Gambar 4.5 menunjukkan respon *tracking* pada *altitude* ketika terdapat penambahan target posisi x dan y .



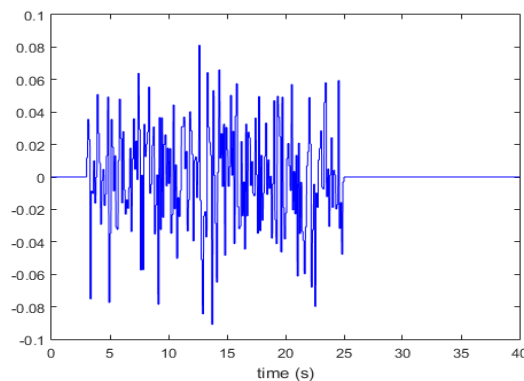
Gambar 4.5 Respon *tracking* pada *altitude* dengan perubahan posisi x dan y .

Dari Gambar 4.5 dapat dilihat bahwa tidak ada perubahan yang signifikan pada respon *altitude*. Respon tetap mengikuti referensi yang diberikan dengan nilai ISE *altitude* sebesar 0.0037 m.

4.2 Simulasi Sistem Kontrol dengan Tambahan Gangguan Terbatas

Untuk menguji kekokohan sistem kontrol *tracking* pada *altitude* yang telah dirancang, dilakukan simulasi dengan penambahan gangguan dan/atau noise. Pada simulasi ini, mekanisme VTOL dilakukan tanpa adanya perubahan posisi x dan y . Sinyal masukan untuk posisi x dan y diatur konstan sama dengan 0 (nol). Hal ini dilakukan untuk menguji kemampuan stabilisasi sistem gerak rotasi terhadap gangguan dan/atau noise yang diberikan.

Pengujian pertama yang dilakukan adalah pemberian noise pada keluaran sudut *roll* dan *pitch*. Noise dibatasi hingga rata-rata amplitudonya adalah 20% dari nilai maksimum sudut *roll* dan *pitch* yang dapat dihadapi oleh quadcopter yaitu 0.5 radian. Pada simulasi ini, noise diberikan pada detik ke-3 sampai dengan detik ke-25. Noise yang diberikan berupa sinyal acak terdistribusi Gaussian dengan *mean* 0 dan *variance* 0.001 sebagaimana ditunjukkan pada Gambar 4.6.

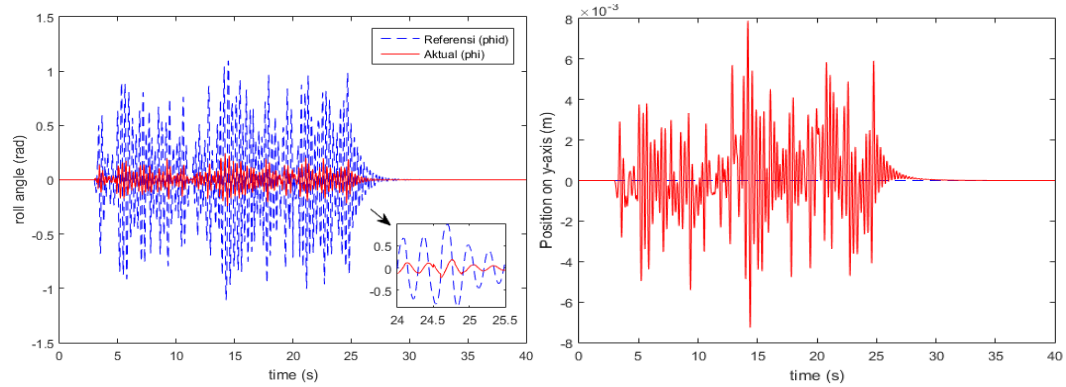


Gambar 4.5 Noise berupa sinyal acak terdistribusi Gaussian dengan *mean* 0 dan *variance* 0.001

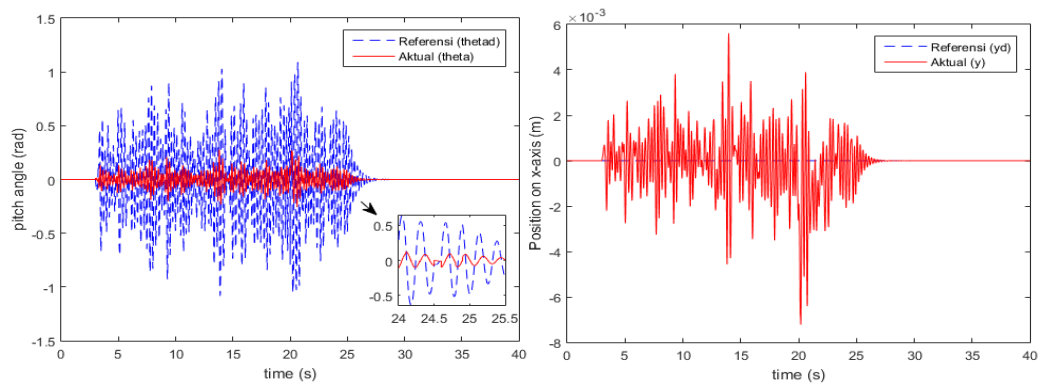
Respon sudut *roll* dan posisi y dapat dilihat pada Gambar 4.6. Pada gambar tersebut diketahui bahwa noise yang diberikan pada sudut *roll* dan *pitch* mengakibatkan posisi y mengalami riak dengan nilai maksimum sebesar 0.0079 m. Begitu juga pada sudut *roll*, terjadi osilasi dengan nilai *undershoot* kurang 0.24 rad.

Respon sudut *pitch* dan posisi x dapat dilihat pada Gambar 4.7. Pada gambar tersebut diketahui bahwa noise yang diberikan pada sudut *roll* dan *pitch* mengakibatkan posisi x mengalami riak kecil dengan nilai maksimum sebesar

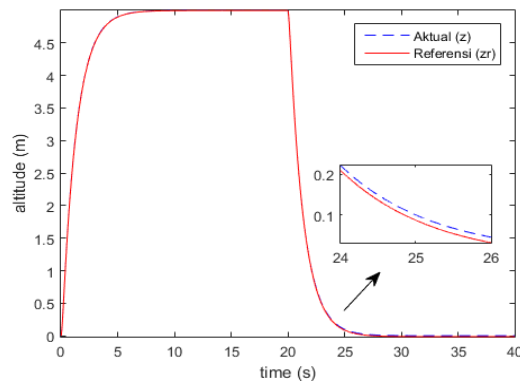
0.0072 m. Begitu juga pada sudut *pitch*, terjadi osilasi dengan nilai *undershoot* kurang 0.278 rad.



Gambar 4.5 Respon sudut *roll* dan posisi *y* ketika diberikan noise pada sudut *roll* dan *pitch*.

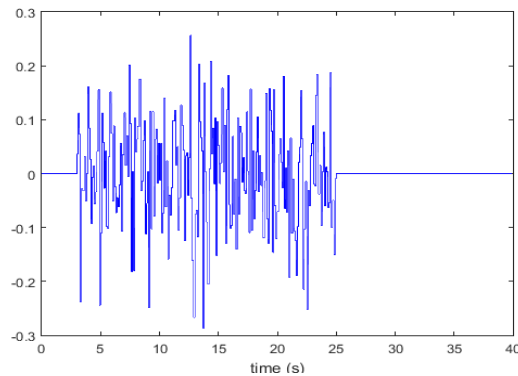


Gambar 4.6 Respon sudut *pitch* dan posisi *x* ketika diberikan noise pada sudut *roll* dan *pitch*.



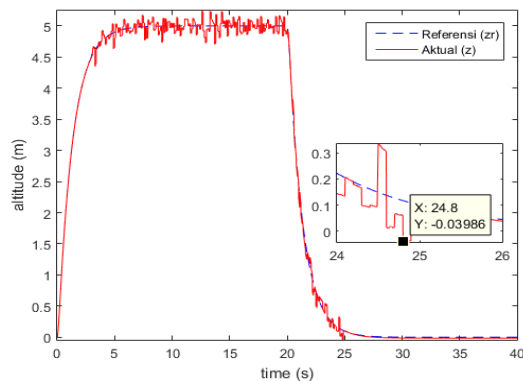
Gambar 4.7 Respon *tracking* pada *altitude* ketika diberikan *noise* pada sudut *roll* dan *pitch*

Noise yang diberikan pada sudut *roll* dan *pitch* tidak memberikan pengaruh yang signifikan terhadap respon *tracking* pada *altitude* sebagaimana dapat dilihat pada Gambar 4.7. Hal ini dibuktikan dengan nilai ISE *altitude* yang diperoleh yaitu 0.0037 m.



Gambar 4.8 Noise berupa sinyal acak terdistribusi Gaussian dengan *mean* 0 dan *variance* 0.01

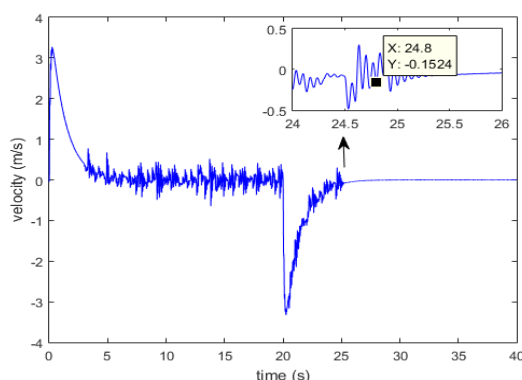
Simulasi selanjutnya adalah simulasi VTOL dengan penambahan *noise* pada keluaran *altitude*. Noise yang digunakan sama dengan *noise* sebelumnya, yaitu berupa *randomnumber* dengan *mean* 0 dan *variance* 0.01 sebagaimana dapat dilihat pada Gambar 4.8. Noise diberikan mulai detik ke-3 sampai detik ke-25.



Gambar 4.9 Respon *tracking* pada *altitude* dengan pemberian noise pada keluaran *altitude*

Pemberian noise pada keluaran *altitude* cukup memberikan pengaruh pada respon *tracking* sehingga terlihat tidak *smooth*. Hal ini dikuatkan dengan nilai ISE *altitude* pada simulasi ini adalah sebesar 0.2278. Pada Gambar 4.9 dapat dilihat

bahwa respon *tracking* pada *altitude* mengalami *drop* dari ketinggian 0.06 m ke titik 0 m pada detik ke 24.8. Jika ditinjau pada respon kecepatan vertikal sebagaimana ditunjukkan Gambar 4.10, diketahui bahwa saat detik ke 24.8 kecepatan vertikal adalah sebesar -0.1524 m/s.



Gambar 4.10. Respon kecepatan vertikal saat diberikan *noise* pada keluaran *altitude*

Percobaan serupa dilakukan dengan parameter *noise* yang berbeda sehingga diketahui nilai ISE *altitude* sebagaimana pada Tabel 4.1. Sistem kontrol *altitude* diuji dengan noise beragam yaitu $N(0, 0.001)$, $N(0, 0.005)$, $N(0, 0.01)$ dan $(0, 0.02)$. Dari hasil simulasi diperoleh bahwa semakin besar nilai *variance* pada *noise* yang diberikan pada keluaran *altitude*, maka nilai ISE *altitude* juga semakin besar. Pada *noise* terbesar yang disimulasikan, yaitu pada *variance* 0.02, terjadi *drop* ketinggian dari 0.4 m ke 0 m. Hal ini cukup membahayakan quadcopter jika terjadi di dunia nyata. Dari simulasi ini diketahui bahwa pada mekanisme VTOL, proses *landing* merupakan proses yang paling kritis terhadap kehadiran *noise*.

Tabel 4.1 Nilai ISE *altitude* pada simulasi dengan berbagai nilai *noise*

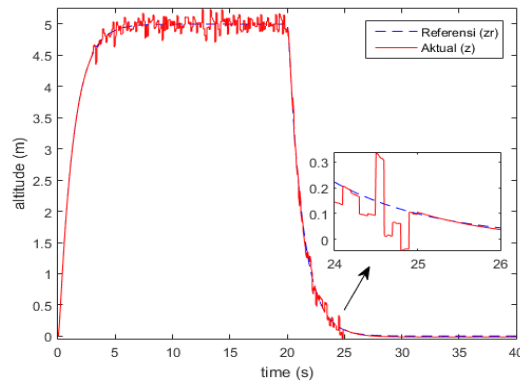
Noise	Nilai ISE Altitude (m)
Mean = 0, Var = 0.001	0.02598
Mean = 0, Var = 0.005	0.1156
Mean = 0, Var = 0.01	0.2278
Mean = 0, Var = 0.02	0.4522

Selanjutnya dilakukan percobaan tambahan untuk simulasi dengan *noise* pada *altitude*, yaitu dengan mengubah nilai level pelemahan ρ . Perubahan nilai ρ akan merubah nilai gain K_i . Pada Tabel 4.2 disajikan nilai K_i yang diperoleh dari perhitungan dengan bantuan MATLAB untuk berbagai nilai ρ . Nilai K_i tersebut kemudian diterapkan pada kontroler untuk dilakukan simulasi.

Tabel 4.2 Daftar nilai K_i untuk berbagai nilai ρ

Nilai level pelemahan (ρ)	Nilai K_i
$\rho = 0.005$	$K_1 = [-16.5699-16.6166]$
	$K_2 = [-16.6579-16.7003]$
	$K_3 = [-16.6579 -16.7003]$
	$K_4 = [-16.7472-16.7853]$
$\rho = 0.008$	$K_1 = [-16.5563-16.6039]$
	$K_2 = [-16.6442-16.6876]$
	$K_3 = [-16.6442 -16.6876]$
	$K_4 = [-16.7334-16.7725]$
$\rho = 0.01$	$K_1 = [-16.5531 -16.6010]$
	$K_2 = [-16.6410 -16.6847]$
	$K_3 = [-16.6410 -16.6847]$
	$K_4 = [-16.7302-16.7696]$

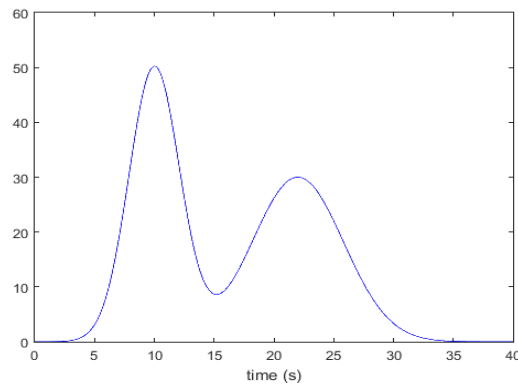
Gambar 4.11 menunjukkan respon *tracking* pada *altitude* dengan *noise* pada keluaran *altitude* untuk $\rho = 0.01$. Tidak ada pengaruh yang signifikan pada respon dengan penggantian nilai ρ . Hal ini terlihat dari nilai ISE *altitude* yang diperoleh terpaut sedikit dengan nilai ISE *altitude* pada nilai ρ sama dengan 0.003 yaitu 0.2276. Begitu juga untuk $\rho = 0.008$ dan $\rho = 0.005$, nilai ISE *altitude* yang diperoleh sama yaitu 0.2276.



Gambar 4.11 Respon *tracking* pada *altitude* dengan *noise* dan $\rho = 0.01$

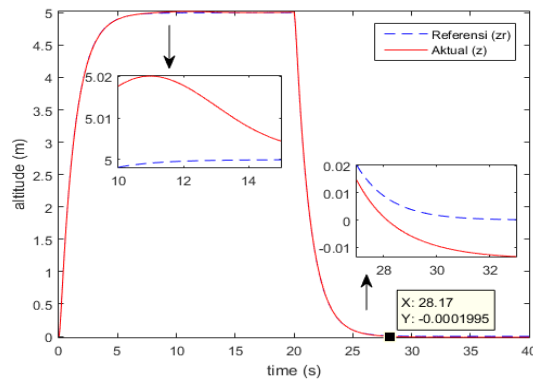
Kemudian sistem kontrol *altitude* diuji dengan penambahan sinyal gangguan berupa angin sebagaimana pada Gambar 4.12. Persamaan 4.1 menunjukkan formulasi sinyal gangguan yang digunakan pada simulasi ini. Sinyal gangguan ini ditambahkan pada dinamika sistem gerak translasi di sumbu *z* (*altitude*).

$$w(x) = k_1 e^{-\left(\frac{x-\mu}{\sigma}\right)^2} + k_2 e^{-\left(\frac{x-\mu}{\sigma}\right)^2} \quad (4.1)$$



Gambar 4.12. Sinyal gangguan

Respon *tracking* pada *altitude* dengan penambahan gangguan angin dapat dilihat pada Gambar 4.13. Pada gambar tersebut terlihat bahwa gangguan menyebabkan terjadinya *overshoot* sekitar 0.4% dari referensi *altitude* ketika proses *take off*. Sedangkan pada proses *landing*, *altitude* sedikit terlambat mencapai titik 0 m yaitu pada detik ke 28.17 dengan kecepatan -0.0089 m/s.



Gambar 4.12 Respon *tracking* pada *altitude* dengan penambahan gangguan

Secara umum, respon menunjukkan bahwa sistem kontrol *altitude* mampu mengatasi gangguan yang diberikan dan mengikuti sinyal referensi yang diberikan. Hal ini ditunjukkan dengan nilai L_2 -gain yang lebih kecil dari nilai level pelemahan ρ yang ditentukan. Nilai L_2 -gain dihitung berdasarkan Persamaan 2.51. Pada simulasi ini, diperoleh nilai L_2 -gain sebesar 0.000539. Sementara nilai ρ yang ditentukan adalah 0.003. Di samping itu, nilai ISE altitude yang terukur sebesar 0.00046. Percobaan pada simulasi ini membuktikan bahwa performa *tracking* H_∞ yang diinginkan telah terpenuhi.

Simulasi dilanjutkan dengan mengganti nilai penguat K_i dari berbagai nilai ρ sebagaimana disajikan pada Tabel 4.2. Nilai L_2 -gain dan ISE pada simulasi ini dapat dilihat di Tabel 4.3. Dari simulasi ini dapat ditarik kesimpulan bahwa semakin kecil nilai level pelemahan, maka nilai L_2 -gain akan semakin kecil, begitu juga dengan nilai ISE. Hal ini sesuai dengan teori performa H_∞ yang dinyatakan dalam Persamaan 2.54.

Tabel 4.3 Hasil simulasi saat kondisi ada gangguan dengan berbagai nilai ρ

Nilai level pelemahan (ρ)	Nilai L_2 -gain	Nilai ISE Altitude (m)
$\rho = 0.003$	0.000539	0.00459
$\rho = 0.005$	0.000558	0.00492
$\rho = 0.008$	0.000564	0.00503
$\rho = 0.01$	0.000566	0.00506

Halaman ini sengaja dikosongkan

BAB 5

PENUTUP

5.1 Kesimpulan

Penelitian ini menyajikan sistem kontrol untuk menyelesaikan permasalahan dalam mekanisme *autonomous VTOL* pada quadcopter dengan menggunakan *input-state feedback linearization* dan fuzzy T-S. Sistem gerak rotasi sebagai *inner-loop* dikontrol menggunakan *input-state feedback linearization* sehingga semua sudut *roll*, *pitch* dan *yaw* dapat stabil di 0 rad. Sedangkan sistem gerak translasi di sumbu *z* dikontrol menggunakan fuzzy T-S sehingga dapat mengikuti sinyal referensi yang diberikan.

Hasil simulasi menunjukkan bahwa sistem kontrol gerak rotasi mampu menstabilkan sudut *roll*, *pitch* dan *yaw* dan respon sesuai karakteristik respon transien yang diinginkan, yaitu *zero offset*, *zero overshoot* dan *settling time* (t_s) 1 detik $\pm 2\%$. Sistem kontrol gerak translasi mampu mengikuti sinyal referensi yang diberikan dan mampu mengatasi gangguan yang diberikan dengan L^2 gain sebesar 0.000539. Hal ini menunjukkan bahwa performa H_∞ dengan level pelemahan sebesar 0,003 telah terpenuhi.

5.2 Saran

Untuk penelitian selanjutnya, pengembangan dapat dilakukan dengan penggabungan sistem kontrol gerak translasi *x*, *y* dan *z* menggunakan metode linierisasi state dependent dengan tetap menggunakan H_∞ sehingga sistem gerak translasi dapat terjamin kekokohnya secara keseluruhan. Kemudian dengan membuat kontrol *tracking* pada sistem gerak rotasi, maka quadcopter dapat melakukan *aerobatic take off and landing*. Selain itu, dapat dipertimbangkan juga mengenai penambahan kontrol orientasi.

Halaman ini sengaja dikosongkan

DAFTAR PUSTAKA

- [1] Bouabdallah, Samir, "Design and Control of Quadrotor with Application to Autonomous Flying," Project report, Ecole Polytechnic, 2007.
- [2] Shahida K., M. Shahid, Ibraheem and Himanshu C., "Dynamic Modeling and Stabilization of Quadrotor Using PID Controller," IEEE Inter. Conf. on Advances in Computing, Comm. and Informatics, pp. 746-750. 2014
- [3] Angelica T.G & Yolanda B., "Modelling & Robust Attitude Control of Quadcopter System," IEEE International Conference on Electrical Engineering, Computing Science and Automatic Control, pp. 7-12, 2013.
- [4] A. Rabhi, M. Chadli and C. Pegard, "Robust Fuzzy Control for Stabilization of A Quadrotor," IEEE International Conference on Advanced Robotics, pp. 471-475, 2011.
- [5] L. Solaque, Zaira P. & M. Doque, "Nonlinear Control of The Airship Cruise Flight Phase with Dynamical Decoupling," IEEE Conference on Electronics, Robotics and Automotive Mechanics, pp. 472-477, 2008.
- [6] A. Benallegue, A. Mokhtari and L. Fridman, Feedback Linearization and High Order Sliding Mode Observer For A Quadrotor UAV, IEEE International Workshop on Variable Structure Systems, pp. 365-372, 2006.
- [7] Zhijun Z., Jianli Y., Yuanking L. and Xiaoyan Z., "A New Neural-Dynamic Control Method of Position and Angular Stabilization for Autonomous Quadrotor UAVs," IEEE International Conference on Fuzzy Systems, pp. 850-855, 2016.
- [8] Tri K.P., Agfianto E.P. and Andi D., "Optimizing Control based on Ant Colony Logic for Quadrotor Stabilization", IEEE Inter. Conf. on Aerospace Electronics and Remote Sensing Tech., pp. 1-4, 2015.
- [9] Hao Liu, Jianxiang Xi and Yisheng Zhong, "Robust Attitude Stabilization for Nonlinear Quadrotor Systems with Uncertainties and Delays," IEEE Trans. on Industrial Electronics, vol. PP, pp. 1-10, 2017.
- [10] David J. Esteves, Alexandra M. and Jose Raul A., "Stabilization and Altitude Control of an Indoor Low-Cost Quadrotor: Design and Experimental Result," IEEE Inter. Conf. on Autonomous Robot Sys. and Competitions, pp. 150-155. 2015.
- [11] W. Hadjadj-Aoul, A. Mokhtari and A. Benallegue, "Asymtotic Stabilization of Quadrotor Helicopter's Attitude Using An Optimal Hierarchical Control Technique," IEEE International Conference on Robotics and Biomimetics, pp. 1709-1713, 2014.

- [12] Wesam Jasim and Dongbing Gu, "H ∞ Control for Quadrotor Attitude Stabilization," IEEE International Conference on Control, pp. 19-24. 2014.
- [13] Nasrettin Koksak, Hao An and Baris Fidan, "Two-Level Nonlinear Tracking Control of a Quadrotor Unmanned Aerial Vehicle," ScienceDirect IFAC-PapersOnline, vol. 49, pp. 254-259, 2016.
- [14] Shane Stebler, William MacKunis and Mahmut Reyhanoglu, "Nonlinear Output Feedback Tracking Control of a Quadrotor UAV in the Presence of Uncertainty," 14th IEEE International Conference on Control, Automation, Robotics and Vision, pp. 1-6, 2016.
- [15] Hyeonbeom Lee and H. Jin Kim, "Robust Control of a Quadrotor using Takagi-Sugeno Fuzzy Model and an LMI Approach," 14th IEEE International Conference on Control, Automation and Systems, pp. 370-374, 2014.
- [16] T. Takagi and M. Sugeno, "Fuzzy Identification of Systems and Its Applications to Modeling and Control," IEEE Transactions on Systems, Man and Cybernetics, Vol. 15, pp. 116-132, 1985.
- [17] Bresciani, Tommaso, "Modelling, Identification and Control of a Quadrotor Helicopter," Departement of Automatic Control Lund University, 2008.
- [18] H. Megawati & Trihastuti A., "Tracking Control Observer-Based using H ∞ Performance with LMI Approach," International Seminar on Intelligent Technology and Its Application, 2014.
- [19] Jeremy G. VanAntwerp and Richard D. Braatz, "A Tutorial on Linear and Bilinear Matrix Inequalities," ScienceDirect Journal of Process Control, vol. 10, pp. 363-385, 2000.
- [20] S. Boyd etc., "Linear Matrix Inequalities in System and Control Theory," Philadelphia, PA:SIAM, 1994.
- [21] Quanser, "Quanser Q-Ball X-4 User Manual," pp. i-46, 2010.
- [22] J. E. Slotine and W. Li, "Applied Nonlinear Control," New Jersey: Prentice Hall International Editions, 1991.
- [23] K. M. Passino and S. Yurkovich, "Fuzzy Control," California: Addison Wesley Longman Inc., 1998.
- [24] S. Kusumadewi & H. Purnomo, "Aplikasi Logika Fuzzy: untuk Pendukung Keputusan," Yogyakarta: Graha Ilmu, 2010.
- [25] K. Ogata, "Modern Control Engineering 3rd Edition," New Jersey: Prentice Hall, 1997.

- [26] H.K. Khalil, "Nonlinear Systems Third Edition," New Jersey: Prentice Hall, 2002.
- [27] F.L. Lewis & V.L. Syrmos, "Optimal Control 2nd Edition," New York: John Wiley & Sons Inc., 1995.
- [28] T. Agustinah, F. Isdaryani & M. Nuh, "Tracking Control of Quadrotor Using Static Output Feedback with Modified Command Generator Tracking," International Review of Automatic Control (IREACO), vol. 9, No. 4, 2016.

Halaman ini sengaja dikosongkan

LAMPIRAN

1. Algoritma *m-file* untuk *membership function*

```
function nthetaphie = fuzzyts(u1, u2)
theta=u1;
phie=u2;
utheta=[0 0]';
uphie=[0 0]';
if theta <-0.174
    utheta(1)=1;
    utheta(2)=0;
elseif theta <0.174
    utheta(1)=(0.174-theta)/(0.348);
    utheta(2)=(theta+0.1)/(0.348);
else
    utheta(1)=0;
    utheta(2)=1;
end
if phie <-0.174
    uphie(1)=1;
    uphie(2)=0;
elseif phie<0.1
    uphie(1)=(0.174-phie)/(0.348);
    uphie(2)=(phie+0.174)/(0.348);
else
    uphie(1)=0;
    uphie(2)=1;
end
%inferensi
thetaphie=[0 0 0 0]';
thetaphie(1)= min(utheta(1),uphie(1));
thetaphie(2)= min(utheta(1),uphie(2));
thetaphie(3)= min(utheta(2),uphie(1));
thetaphie(4)= min(utheta(2),uphie(2));
thetaphie=[thetaphie(1) thetaphie(2) thetaphie(3)
thetaphie(4)]'
%normalisasi
nthetaphie=thetaphie/sum(thetaphie)
```

2. Algoritma LMI untuk menghitung K_i dan P_i

```
clear all
clc
A=[0 1;0 0];
B1=[0;1/3.5];
B2=[0;0.985/3.5];
B3=[0;0.985/3.5];
B4=[0;0.970/3.5];
Ar=[0 1;-2.28 -4.57];
Br=[0;2.28];
C=[1 0;0 1];
Q=[44*10^-10 0;0 37*10^-10];

qq=inv(Q);
aa=0.01;
```

```

ab=aa^2;
bb=1/(aa^2);

setlmis([]);
W11=lmivar(1,[2 1]);
Y1=lmivar(2,[1 2]);
Y2=lmivar(2,[1 2]);
Y3=lmivar(2,[1 2]);
Y4=lmivar(2,[1 2]);

lmiterm([1 1 1 W11],1,A,'s');
lmiterm([1 1 1 Y1],B1,1,'s');
lmiterm([1 1 1 0],bb);
lmiterm([1 2 1 W11],1,1);
lmiterm([1 2 2 0],-qq);

lmiterm([2 1 1 W11],1,A,'s');
lmiterm([2 1 1 Y2],B2,1,'s');
lmiterm([2 1 1 0],bb);
lmiterm([2 2 1 W11],1,1);
lmiterm([2 2 2 0],-qq);

lmiterm([3 1 1 W11],1,A,'s');
lmiterm([3 1 1 Y3],B3,1,'s');
lmiterm([3 1 1 0],bb);
lmiterm([3 2 1 W11],1,1);
lmiterm([3 2 2 0],-qq);

lmiterm([4 1 1 W11],1,A,'s');
lmiterm([4 1 1 Y4],B4,1,'s');
lmiterm([4 1 1 0],bb);
lmiterm([4 2 1 W11],1,1);
lmiterm([4 2 2 0],-qq);

lmiterm([-5 1 1 W11],1,1);

lmis=getlmis;
[tmin,xfeas]=feasp(lmis);
w11=dec2mat(lmis,xfeas,W11);
y1=dec2mat(lmis,xfeas,Y1);
y2=dec2mat(lmis,xfeas,Y2);
y3=dec2mat(lmis,xfeas,Y3);
y4=dec2mat(lmis,xfeas,Y4);

evlmi = evallmi(lmis,xfeas);
[lhs1,rhs1]=showlmi(evlmi,1);

W11e=w11;
Ye1=y1;
Ye2=y2;
Ye3=y3;
Ye4=y4;
P11=inv(w11)
K1=y1*inv(w11)
K2=y2*inv(w11)
K3=y3*inv(w11)

```

```

K4=y4*inv(w11)

setlmis([]);
P22=lmivar(1,[2 1]);

lmiterm([1 1 1 0],(A+B1*K1)'*P11+P11*(A+B1*K1)+bb*P11*P11+Q);
lmiterm([1 1 2 0],((-1)*(P11*B1*K1)-Q));
lmiterm([1 1 3 0],0);
lmiterm([1 2 1 0],((-1)*(P11*B1*K1)-Q)');
lmiterm([1 2 2 P22],Ar',1,'s');
lmiterm([1 2 2 0],Q);
lmiterm([1 2 3 P22],1,Br,1);
lmiterm([1 3 1 0],0);
lmiterm([1 3 2 P22],Br',1);
lmiterm([1 3 3 0],-ab);

lmiterm([2 1 1 0],(A+B2*K2)'*P11+P11*(A+B2*K2)+bb*P11*P11+Q);
lmiterm([2 1 2 0],((-1)*(P11*B2*K2)-Q));
lmiterm([2 1 3 0],0);
lmiterm([2 2 1 0],((-1)*(P11*B2*K2)-Q)');
lmiterm([2 2 2 P22],Ar',1,'s');
lmiterm([2 2 2 0],Q);
lmiterm([2 3 1 0],0);
lmiterm([2 2 3 P22],1,Br,1);
lmiterm([2 3 2 P22],Br',1);
lmiterm([2 3 3 0],-ab);

lmiterm([3 1 1 0],(A+B3*K3)'*P11+P11*(A+B3*K3)+bb*P11*P11+Q);
lmiterm([3 1 2 0],((-1)*(P11*B3*K3)-Q));
lmiterm([3 1 3 0],0);
lmiterm([3 2 1 0],((-1)*(P11*B3*K3)-Q)');
lmiterm([3 2 2 P22],Ar',1,'s');
lmiterm([3 2 2 0],Q);
lmiterm([3 2 3 P22],1,Br,1);
lmiterm([3 3 1 0],0);
lmiterm([3 3 2 P22],Br',1);
lmiterm([3 3 3 0],-ab);

lmiterm([4 1 1 0],(A+B4*K4)'*P11+P11*(A+B4*K4)+bb*P11*P11+Q);
lmiterm([4 1 2 0],((-1)*(P11*B4*K4)-Q));
lmiterm([4 1 3 0],0);
lmiterm([4 2 1 0],((-1)*(P11*B4*K4)-Q)');
lmiterm([4 2 2 P22],Ar',1,'s');
lmiterm([4 2 2 0],Q);
lmiterm([4 2 3 P22],1,Br,1);
lmiterm([4 3 1 0],0);
lmiterm([4 3 2 P22],Br',1);
lmiterm([4 3 3 0],-ab);

lmiterm([-5 1 1 P22],1,1);

lmis=getlmis;
[tmin,xfeas]=feasp(lmis);
P22=dec2mat(lmis,xfeas,P22)

evlmi = evallmi(lmis,xfeas);

```

```
[lhs1,rhs1]=showlmi (evlmi,1);
```


Riwayat Penulis



Chalidia Nurin Hamdani dilahirkan pada Tahun 1990 di Sumenep, Jawa Timur. Penulis menempuh pendidikan tingginya di Jurusan Teknik Elektro, Institut Teknologi Sepuluh Nopember pada Tahun 2009 dan mendapatkan gelar sarjana pada Tahun 2013. Kemudian penulis melanjutkan pendidikan di Program Magister Bidang Keahlian Teknik Sistem Pengaturan, Jurusan Teknik Elektro ITS dengan bantuan beasiswa *fresh graduate* dari Departemen Pendidikan Tinggi (DIKTI). Saat ini Juli 2017, selain sedang menyelesaikan studi program magisternya, penulis juga bekerja sebagai Pengawas Konservasi Energi di Direktorat Jenderal Energi Baru, Terbarukan dan Konservasi, Kementerian Energi dan Sumber Daya Mineral Republik Indonesia.